THE DESIGN AND SIMULATION ANALYSIS OF AN ATTITUDE DETERMINATION AND CONTROL SYSTEM FOR A SMALL EARTH OBSERVATION SATELLITE

by

Gerhard Hermann Janse van Vuuren

Thesis presented in partial fulfilment of the requirements for the degree of
Master of Engineering at Stellenbosch University

Supervisor:

Prof WH Steyn

Department of Electrical and Electronic Engineering

March 2015

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the owner of the copyright thereof (unless to the extent explicitly otherwise stated) and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: March 2015

Abstract

The ability of satellites to actively control their attitude has changed the way we live. Navigation systems, satellite television, and weather forecasting, for example, all rely on satellites which are able to determine and control their attitude accurately.

This project was aimed at designing and analysing an attitude determination and control system (ADCS) for a 20 kg Earth observation satellite by means of simulation. A realistic simulation toolset, which includes the space environment, sensor, and actuator models, was created using MATLAB and Simulink. An ADCS hardware suite was selected for the satellite based on a given set of pointing and stability requirements, as well as current trends in the small satellite industry. The hardware suite consists of among others a star tracker and three reaction wheels.

A variety of estimators and controllers were investigated, after which an application specific ADCS state machine was defined. The state machine included a Safe Mode for de-tumbling, a Nominal Mode for normal operation, a Forward Motion Compensation (FMC) Imaging Mode for Earth observation, and a Target Tracking Mode for ground station tracking. Simulation results indicated that de-tumbling, coarse and fine sun tracking, FMC factor 4 imaging, and target tracking were successfully implemented. Lastly, the satellite's pointing error and stability were determined to be less than 70 arcseconds and 7 arcseconds per second respectively, both values well within the given requirements.

Samevatting

Satelliete se vermoë om hul oriëntasie aktief te beheer, het die manier waarop ons lewe, verander. Navigasiestelsels, satelliettelevisie en weervoorspelling, byvoorbeeld, maak staat op satelliete wat hul oriëntasie akkuraat kan bepaal en beheer.

Die mikpunt van hierdie projek was die ontwerp en analise van 'n oriëntasiebepaling- en -beheerstelsel (ADCS) vir 'n 20 kg aardwaarnemingsatelliet deur middel van simulasie. 'n Realistiese simulasie-opstelling, wat modelle van die ruimteomgewing, sensore en aktueerders insluit, was ontwikkel deur gebruik te maak van MATLAB en Simulink. 'n ADCS hardewarestel was gekies vir die satelliet op grond van 'n stel rig- en stabiliteitsvereistes, sowel as die huidige tendense in die klein-satellietbedryf. Die hardewarestel bestaan onder andere uit 'n stervolger en drie reaksiewiele.

Nadat verskeie afskatters en beheerders ondersoek was, was 'n toepassingspesifieke ADCS toestand-masjien gedefinieer. Die toestandmasjien het 'n Veilige Modus vir onttuimelling, 'n Nominale Modus vir normale operasie, 'n Vorentoe-bewegingskompensering (FMC) Beeldskandeermodus vir aardwaar-neming en 'n Teikenvolgmodus vir grondstasie volging ingesluit. Simulasieresultate het aangedui dat onttuimeling, growwe- en fyn sonvolging, FMC faktor 4 beeldskandering en teikenvolging suksesvol geïmplementeer was. Laastens was die satelliet se rigfout en stabiliteit bepaal as minder as 70 boogs-ekondes en 7 boogsekondes per sekonde onderskeidelik, albei waardes gemaklik binne die vereistes.

Acknowledgements

I would like to extend my sincere gratitude to the following persons:

- Professor WH Steyn, for his guidance and for his willingness to share his seemingly endless knowledge.

- My ESL colleagues, in particular André Heunis, Jan-Hielke le Roux, Willem Jordaan, Mike-Alec Kearney, Mohammed Bin Othman, and Muhammad Junaid, for the numerous discussions and the camaraderie.

- Carla, my wife, for her unconditional love and support.

Contents

List of Figures

List of Tables

List of Abbreviations and Acronyms

ADCS   Attitude Determination and Control System

AIS   Automatic Identification System

AP   Argument of Perigee

AU   Astronomical Units

CoM   Centre of Mass

CoP   Centre of Pressure

COTS   Commercial-Off-The-Shelf

CSS   Coarse Sun Sensor

D/A   Digital-to-Analogue

DCM   Direction Cosine Matrix

EKF   Extended Kalman Filter

EO   Earth Observation

ESL   Electronic Systems Laboratory

FMC   Forward Motion Compensation

FOV   Field Of View

FSS   Fine Sun Sensor

G/S   Ground Station

GHA   Greenwich Hour Angle

GPS   Global Positioning System

GSD   Ground Sampling Distance

IC   Integrated Circuit

IAGA   International Association of Geomagnetism and Aeronomy

IGRF   International Geomagnetic Reference Field

JD   Julian Date

KF	Kalman Filter
LEO	Low Earth Orbit
LTAN	Local Time of the Ascending Node
MEMS	Micro-ElectroMechanical Systems
MoI	Moment of Inertia
NORAD	North American Aerospace Defence Command
PI	Proportional-Integral
PID	Proportional-Integral-Derivative
PWM	Pulse Width Modulation
RAAN	Right Ascension of the Ascending Node
RKF	Rate Kalman Filter
RMS	Root-Mean-Square
RW	Reaction Wheel
RPY	Roll-Pitch-Yaw
SGP	Simplified General Perturbations
SGP4	Simplified General Perturbations 4
SRP	Solar Radiation Pressure
SSO	Sun-Synchronous Orbit
STR#3	SpaceTrack Report number 3
TLE	Two-Line Elements
ZOH	Zero Order Hold

Chapter 1

Introduction

1.1   Problem Statement

Satellite attitude determination and attitude control play a fundamental role in almost every single satellite mission. The ability of a satellite to actively control its attitude enables the use of a wide variety of payloads. The attitude determination and control system (ADCS) of a satellite is furthermore responsible for de-tumbling the satellite after release, a process without which most satellites would be deemed useless.

The tremendous growth in the small satellite industry over the past decade has enabled the development of small satellite subsystems of which the performance is comparable to that of larger satellites. As a result, Earth observation (EO) has become a feasible mission goal for small satellites. However, an EO satellite still requires a stable platform and the ability to perform high accuracy pointing. The need thus arises for the development of an ADCS to meet the above-mentioned requirements.

The pre-flight performance of an ADCS can furthermore only be determined through simulation. A realistic simulation environment is consequently required with which the correct functioning of the ADCS can be verified. The project at hand was therefore aimed at investigating the development of an ADCS for a 20 kg Earth observation satellite, as well as the development of a simulation environment with which the performance of the ADCS can be analysed.

1.2   Background

The field of ADCS has expanded by orders of magnitude since the launch of Sputnik 1 on 4 October 1957. In just over 50 years, ADCSs have gone from simple spin stabilisation methods to complex optimal control strategies using highly powerful on-board computers. These advances have enabled the use of among others the Global Positioning System (GPS), satellite television, and Google Maps, all of which have become an integral part of the daily lives of people all around the world.

An ADCS consists of both hardware and software, both being equally important in the process of controlling of a satellite. Although there is a vast amount of ADCS software techniques in existence, the number of different ADCS actuators is quite limited. The selection of an ADCS hardware suite for a satellite is heavily dependent on the primary mission payload, but it is also affected by constraints such as volume, mass, and power.

An investigation into the ADCSs of various small satellites (ranging from 6.5 kg to 94 kg in mass) launched over the past 25 years was undertaken as part of this project to determine the current trend in the use of small satellites with specific emphasis on ADCS hardware. The database of the 42 satellites that were included in the investigation can be seen in Appendix A. It should be noted that only satellites on which ADCS data could be found were included.

The first trend that was observed during the investigation was the exponential increase in the amount of small satellites launched. Figure 1.1 shows the launch dates of the satellites included in the database. A trend-line has also been added to emphasise the exponential increase. Had the CubeSat industry been included in this investigation, the exponential trend in Figure 1.1 would most likely have been of a much higher order.
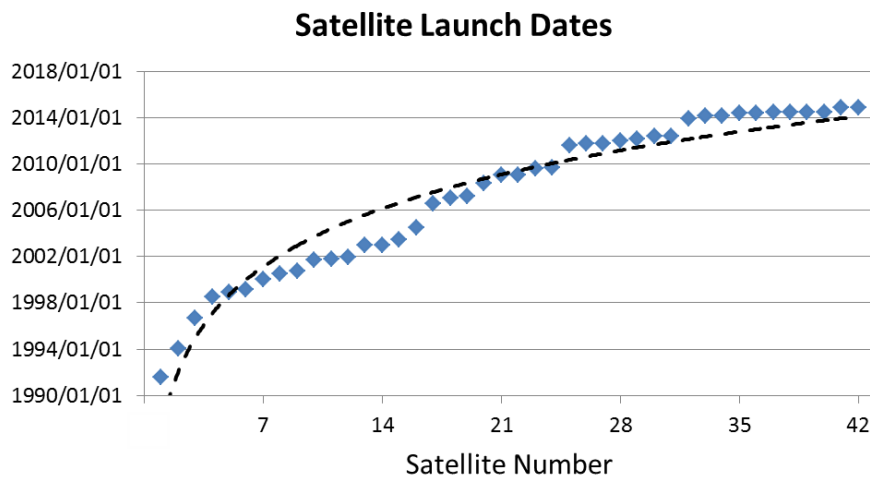
Figure 1.1 – The exponential increase in the number of small satellites launched over the past 25 years

Another observation made during the investigation was that almost every satellite had some form of attitude control system, whether it was passive or active. Active control implies the use of magnetic torquers, momentum or reaction wheels, propulsion systems, control moment gyros, and so forth, whereas passive control includes among others permanent magnets and gravity gradient booms. Figure 1.2(a) indicates the use of attitude control on the investigated satellites. The satellites that did not employ active attitude control were mostly technology demonstration satellites, whereas the primary mission goals of the actively controlled satellites included Earth observation, communication,

science research, technology demonstration, and Automatic Identification System (AIS). Figure 1.2(b) illustrates the various mission goals of the actively controlled satellites.
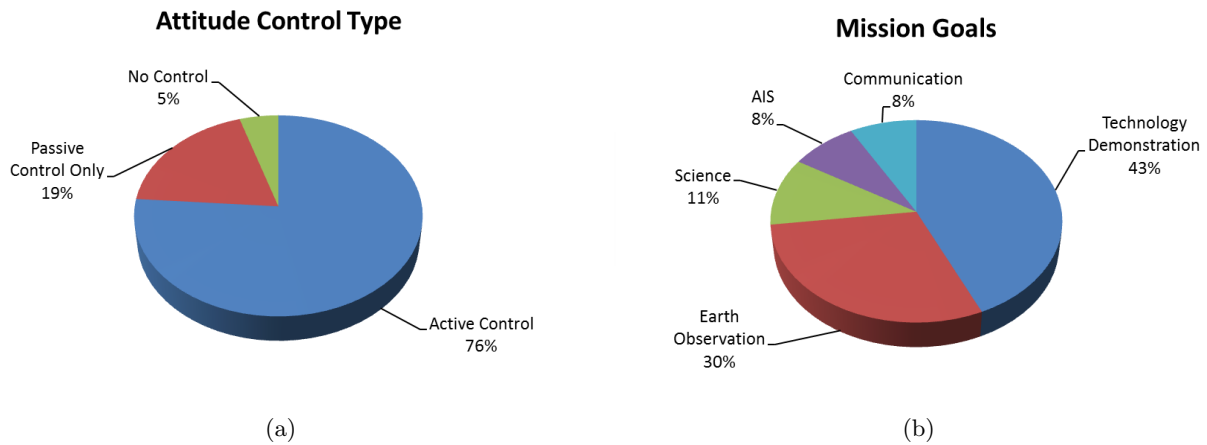
Figure 1.2 – The attitude control types and mission goals of small satellites

It was furthermore found that magnetometers and sun sensors were the most common sensor type, whereas earth/horizon sensors were the least used sensor type. Only a third of the satellites made use of a GPS, whereas the other satellites most likely relied on software orbit propagation techniques. The addition of a GPS will improve the performance of the ADCS, since environment models such as the sun orbit model rely on the current position of the satellite. An error in estimated position will thus translate to an error in attitude estimation. Five of the 33 satellites that used sun sensors made use of solar panels to measure the sun vector. This approach is only feasible for 3-axis measurements if a sufficient amount of solar panels are present. Figure 1.3 illustrates the use of ADCS sensors on-board the satellites that employed attitude control.
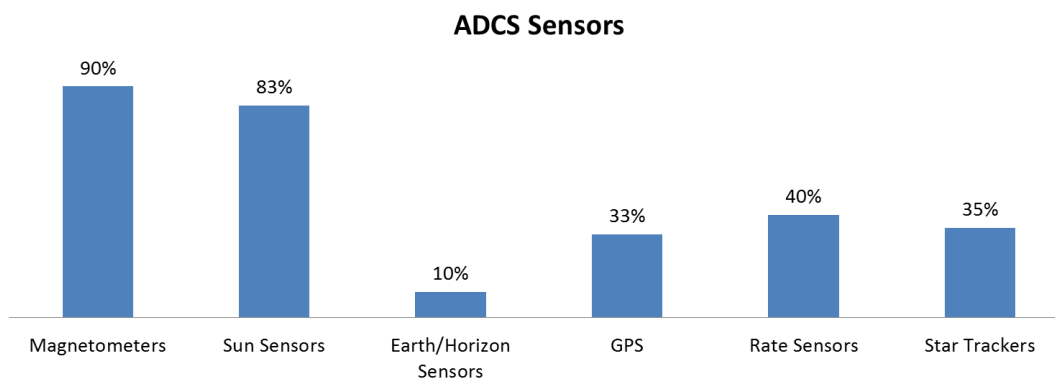
Figure 1.3 – The use of ADCS sensors on small satellites

An interesting finding was that every satellite that was controlled, utilised magnetic control. A fifth of the satellites used permanent magnets (i.e. passive magnetic control), whereas magnetic torquers were

used on all the other satellites. Reaction wheels were also found to be quite common on the satellites. One satellite made use of a control moment gyro to perform attitude control. The aforementioned satellite, Tsubame, was launched on 6 November 2014, which emphasises the fact that ADCS modules traditionally used on larger satellites (such as control moment gyros) are being miniaturised for use on smaller satellites as a result of the exponential growth rate of the small satellite industry. The use of ADCS actuators on the attitude-controlled satellites can be seen in Figure 1.4.
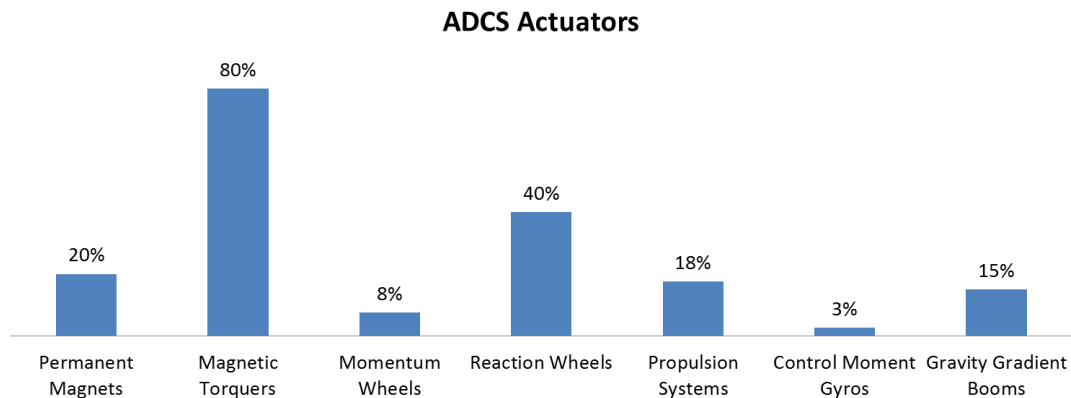
Figure 1.4 – The use of ADCS actuators on small satellites

The research investigation revealed that almost a third of the satellites utilised some form of redundancy, either for a sensor or for an actuator. Redundant magnetometers were present mainly on satellites with a mass less than 15 kg, whereas the average mass of satellites that employed reaction wheel redundancy was calculated as 58 kg. A well known drawback of smaller satellites is their susceptibility to radiation, which leads to the need for redundancy. Conversely, the limitations in terms of budget, power, volume, and mass discourages redundancy. A trade-off between redundancy and one or more of the above-mentioned limitations must therefore often be made in the process of developing a small satellite.

Earth observation satellites were generally found to be larger than the other types of satellites, with an average satellite mass of 54 kg (compared to the average satellite mass of the entire database of 33 kg). Only one of the EO satellites did not employ a star tracker and only one EO satellite did not utilise reaction wheels or control moment gyros for attitude control. **The general trend seems to be that an Earth observation satellite makes use of a magnetometer, sun sensors, a star tracker, magnetic torquers, and reaction wheels for attitude determination and control**.

It was lastly found that for satellites with a mass less than 30 kg, only 14% employed reaction wheels and none of the satellites made use of a star tracker. **These findings clearly indicate that the use of highly accurate sensors and precision pointing actuators on small satellites needs to be investigated.**

1.3   ADCS Concepts

A proper understanding of several fundamental ADCS concepts is crucial in the process of planning a satellite mission. The majority of these concepts are embedded deeply in the ADCS software on-board satellites. The concepts that were used extensively throughout the course of this project will be discussed in this section.

1.3.1   Coordinate Frames

A coordinate frame can be defined by a set of three orthogonal unit vectors, often referred to as $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$. These three vectors are also frequently called the X-axis, Y-axis and Z-axis in space-related literature. An arbitrary vector $\mathbf{w}$ can be expressed in terms of $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ as

$$\mathbf{w} \;\; = \;\; w_x\mathbf{x} + w_y\mathbf{y} + w_z\mathbf{z} \; , \tag{1.3.1}$$

where w_x, w_y, and w_z are the components of $\mathbf{w}$ in the respective directions of $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$. The centre (or the origin) of the three orthogonal vectors is also an important factor in the process of defining the coordinate frame.

The centre of an ADCS coordinate frame is either the centre of the Earth or the centre of the satellite. Two **Earth-centred** coordinate frames were used during this project, namely the Earth fixed coordinate frame (EFC) and the Earth inertial coordinate frame (EIC). As the names suggest, EFC is fixed to the Earth (and thus rotates with it), whereas EIC is inertial-fixed.

The Z-axis of the EFC frame is in the direction of the North Pole. The X-axis is towards the equator at zero longitude and the Y-axis completes the orthogonal set. Although the Z-axis of EIC is the same as that of EFC, the X-axis of EIC is in the direction of Υ, the Vernal Equinox. The Y-axis once again completes the orthogonal set. Given that the Z-axes of the two frames are equal, the EFC frame can be seen as a rotating EIC frame. The angle by which the EFC has been rotated is the Greenwich Hour Angle (GHA), or α_G. Figure 1.5 illustrates this relationship by showing both Earth-centred frames.

A vector in EFC (e.g. $\mathbf{x}$) can be transformed to a vector in EIC (e.g. $\mathbf{y}$) using

$$\mathbf{y} = \mathbf{A}_{EFC}^{EIC}\mathbf{x} \; , \tag{1.3.2}$$

where $\mathbf{A}_{EFC}^{EIC}$ is the EFC-to-EIC transformation matrix, which is dependent on the rotation angle α_G. If $\alpha_{G,0}$ is the GHA at $t = t_0$, then $\mathbf{A}_{EFC}^{EIC}$ can be expressed as

$$\mathbf{A}_{EFC}^{EIC} \;\; = \;\; \begin{bmatrix} \cos(\omega_E t + \alpha_{G,0}) & -\sin(\omega_E t + \alpha_{G,0}) & 0 \\ \sin(\omega_E t + \alpha_{G,0}) & \cos(\omega_E t + \alpha_{G,0}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \; , \tag{1.3.3}$$
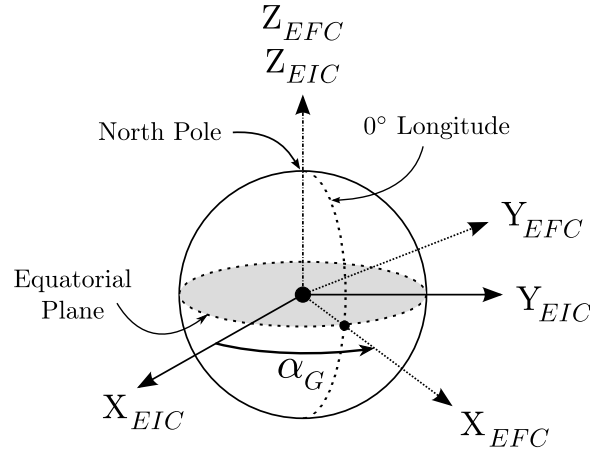
Figure 1.5 – An illustration of the EFC and EIC Earth-centred frames

where ω_E is the rotation rate of the Earth and t is the time elapsed since t_0. α_G can thus be expressed as

$$\alpha_G = \omega_E t + \alpha_{G,0} . \tag{1.3.4}$$

Three **satellite-centred** coordinate frames were used during the course of this project, namely the inertial-referenced coordinate (IRC) frame, the orbit-referenced coordinate (ORC) frame, and the satellite body coordinate (SBC) frame.

The **IRC** frame remains inertial-fixed (as the name suggests) throughout the satellite's orbit. The Z-axis (Z_I) is defined to be towards the centre of the Earth when the satellite is at perigee (the point in the orbit when the satellite is closest to the Earth). The Y-axis (Y_I) is in the opposite direction of the orbit normal vector and the X-axis (X_I) completes the orthogonal set.

ORC is a frame that rotates slowly throughout the orbit. The Z-axis (Z_O) is directed towards the centre of the Earth (i.e. the nadir direction) at all times. The Y-axis (Y_O) is equal to the orbit anti-normal and the X-axis (X_O) completes the orthogonal set (as with the IRC frame). It should be noted that for circular orbits, X_O is in the direction of the velocity vector. Figure 1.6 illustrates the IRC and ORC frames.

The transformation from EIC to ORC is also significant in the process of attitude determination and control. The transformation matrix $\mathbf{A}_{EIC}^{ORC}$ is a function of the satellite's unit position vector $\mathbf{u}$ and unit velocity vector $\mathbf{v}$, both in EIC [1], and can be expressed as

$$\mathbf{A}_{EIC}^{ORC} = \begin{bmatrix} \mathbf{a} & \mathbf{b} & \mathbf{c} \end{bmatrix}^T , \tag{1.3.5}$$

$$\text{where} \quad \mathbf{c} = -\mathbf{u} , \tag{1.3.6}$$

$$\mathbf{b} = \frac{\mathbf{v} \times \mathbf{u}}{\|\mathbf{v} \times \mathbf{u}\|} , \tag{1.3.7}$$

$$\text{and} \quad \mathbf{a} = \mathbf{b} \times \mathbf{c} . \tag{1.3.8}$$

(a) IRC

(b) ORC
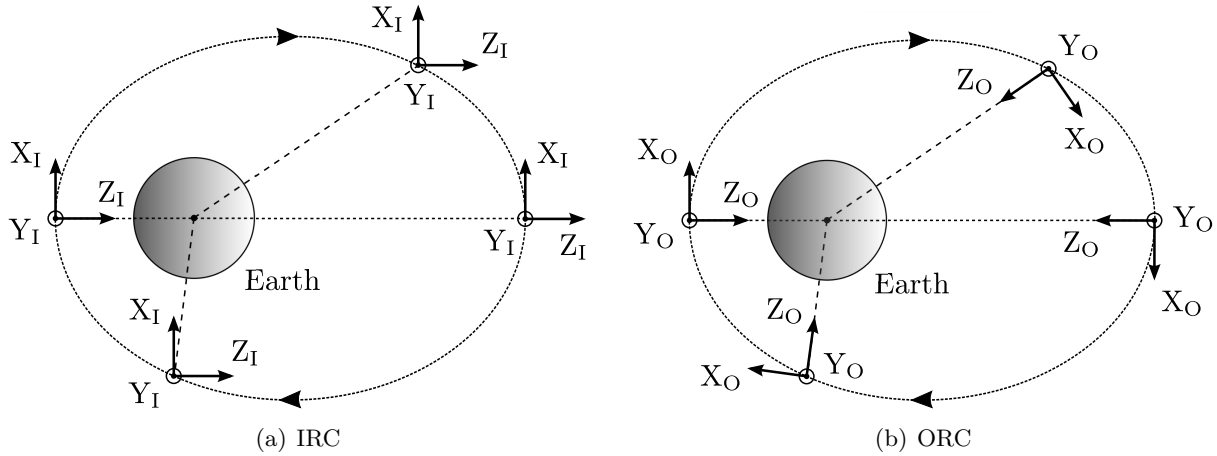
Figure 1.6 – An illustration of the IRC and ORC satellite-centred frames

A physical interpretation of Equation 1.3.5 can be constructed by examining the relationship between the ORC frame and **u** and **v**. The ORC Z-axis is in exactly the opposite direction as **u**, which explains Equation 1.3.6. Given that **u** and **v** are in the orbit plane, the ORC Y-axis is perpendicular to both these vectors, hence the cross-product in Equation 1.3.7. The ORC X-axis completes the orthogonal set, as is the case with the vector **a** in Equation 1.3.8.

The **SBC** frame is fixed to the satellite body (and thus rotates with it). It is often chosen that when the satellite is in a nominal nadir-pointing position, the SBC frame coincides with the ORC frame. For a circular orbit, this would imply that the body Z-axis Z_B is in the nadir direction, the Y-axis Y_B is in the direction of the orbit anti-normal, and the X-axis X_B is in the direction of velocity. A vector in ORC (e.g. **x**) can be transformed to a vector in SBC (e.g. **y**) using

$$\mathbf{y} \;=\; \mathbf{A}_{ORC}^{SBC}\mathbf{x} \;, \tag{1.3.9}$$

where $\mathbf{A}_{ORC}^{SBC}$ (or just **A** for short-hand representation) is the ORC-to-SBC transformation matrix. **A** is also known as the direction cosine matrix (DCM) and it represents the current orientation (or attitude) of the satellite with respect to the ORC frame. The determination of **A** is one of the main components of ADCS that was investigated during the course of this project.

1.3.2 Euler Angles

The DCM that represents the attitude of a satellite can be constructed from three successive Euler rotations about the satellite body axes. The rotation angles about X_B, Y_B, and Z_B are referred to as the roll (ϕ), pitch (θ), and yaw (ψ) angles respectively. The sequence in which the rotations are performed, for example yaw-roll-pitch, is also significant. The Euler-213 sequence (or the pitch-roll-yaw sequence) was chosen for this project and an illustration thereof can be seen in Figure 1.7.

(a) Pitch rotation   (b) Roll rotation   (c) Yaw rotation
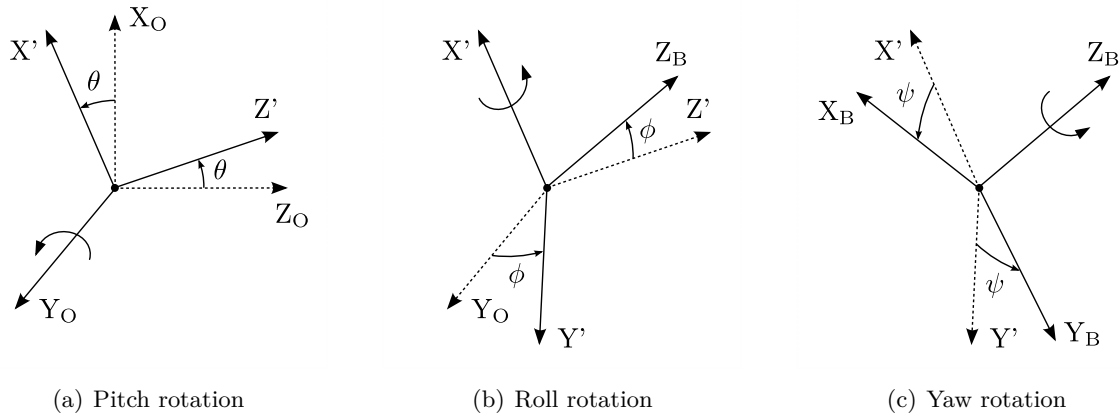
Figure 1.7 – The Euler-213 rotation sequence

Each rotation in Figure 1.7 can be expressed mathematically through a transformation matrix. If $\mathbf{A}_\theta$, $\mathbf{A}_\phi$, and $\mathbf{A}_\psi$ represent the respective transformation matrices of the pitch, roll, and yaw rotations, then the Euler-213 direction cosine matrix $\mathbf{A}$ is given by

$$
\begin{aligned}
\mathbf{A} \;=\; & \mathbf{A}_\psi \mathbf{A}_\phi \mathbf{A}_\theta \\[2mm]
=\; & \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \\[2mm]
=\; & \begin{bmatrix} \cos\theta\cos\psi + \sin\theta\sin\phi\sin\psi & \cos\phi\sin\psi & -\sin\theta\cos\psi + \cos\theta\sin\phi\sin\psi \\ -\cos\theta\sin\psi + \sin\theta\sin\phi\cos\psi & \cos\phi\cos\psi & \sin\theta\sin\psi + \cos\theta\sin\phi\cos\psi \\ \sin\theta\cos\phi & -\sin\phi & \cos\theta\cos\phi \end{bmatrix}
\end{aligned}
\tag{1.3.10}
$$

If a_{ij} represents the elements of $\mathbf{A}$, then the Euler angles can be expressed as a function of $\mathbf{A}$ via

$$
\phi \;=\; -\mathrm{asin}\,(a_{32}) \;, \tag{1.3.11}
$$

$$
\theta \;=\; \mathrm{atan2}\left(\frac{a_{31}}{a_{33}}\right) \;, \tag{1.3.12}
$$

$$
\text{and} \quad \psi \;=\; \mathrm{atan2}\left(\frac{a_{12}}{a_{22}}\right) \;. \tag{1.3.13}
$$

1.3.3   Quaternions

Attitude representation through Euler angles or direction cosine matrices poses several disadvantages. Although Euler angles provide a clear physical interpretation of attitude, trigonometric functions need to be performed and singularities (i.e. no solution) can occur at certain angles. The DCM on the other hand contains no trigonometric functions, but six redundant parameters must be calculated.

Euler symmetric parameters (or quaternions) can also be used to represent a satellite's attitude. Quaternions have been described as "more convenient" than the Euler angles and "more compact" than the DCM [2], since no singularities can occur and only four parameters need to be calculated. However, quaternions do not provide a physical interpretation of attitude.

Quaternions are defined by the parameters of an Euler rotation. If the axis of the Euler rotation from ORC to SBC is defined by the unit vector $\mathbf{e} = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}^T$ and the angle of the Euler rotation is Φ, then the attitude quaternion is defined as

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} e_1 \sin\left(\frac{\Phi}{2}\right) \\ e_2 \sin\left(\frac{\Phi}{2}\right) \\ e_3 \sin\left(\frac{\Phi}{2}\right) \\ \cos\left(\frac{\Phi}{2}\right) \end{bmatrix} . \tag{1.3.14}$$

The components of $\mathbf{e}$ have the special characteristic that they retain their magnitude from the ORC frame to the SBC frame. The quaternion components are not independent of each other, but rather constrained through

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 . \tag{1.3.15}$$

It can also be shown [2] that $\mathbf{A}$ is a function of $\mathbf{q}$ via

$$\mathbf{A} = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_2q_3 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} . \tag{1.3.16}$$

The quaternion components can conversely be expressed in terms of the elements of $\mathbf{A}$ as

$$q_4 = \pm\frac{1}{2}\sqrt{1 + a_{11} + a_{22} + a_{33}} , \tag{1.3.17}$$

$$q_1 = \frac{1}{4q_4}(a_{23} - a_{32}) , \tag{1.3.18}$$

$$q_2 = \frac{1}{4q_4}(a_{31} - a_{13}) , \tag{1.3.19}$$

$$\text{and} \quad q_3 = \frac{1}{4q_4}(a_{12} - a_{21}) . \tag{1.3.20}$$

The sign ambiguity in Equation 1.3.17 indicates that $\mathbf{q}$ and $-\mathbf{q}$ represent exactly the same attitude. Any one of q_1, q_2, and q_3 can also be used to calculate the remaining three quaternion components by changing the appropriate signs of a_{11}, a_{22}, and a_{33} in Equation 1.3.17. "Numerical inaccuracy can be minimised" by using the largest of the four quaternion components to calculate the remaining three [2].

The difference between two satellite attitudes (e.g. a current attitude **q** and a commanded attitude **q**$_c$) can furthermore be expressed as a quaternion error **q**$_{err}$. The quaternion error is calculated by taking the quaternion difference (denoted by the operator $\ominus$) between **q** and **q**$_c$, thus

$$\mathbf{q}_{err} = \mathbf{q} \ominus \mathbf{q}_c \tag{1.3.21}$$

$$\implies \begin{bmatrix} q_{err,1} \\ q_{err,2} \\ q_{err,3} \\ q_{err,4} \end{bmatrix} = \begin{bmatrix} q_{c4} & q_{c3} & -q_{c2} & -q_{c1} \\ -q_{c3} & q_{c4} & q_{c1} & -q_{c2} \\ q_{c2} & -q_{c1} & q_{c4} & -q_{c3} \\ q_{c1} & q_{c2} & q_{c3} & q_{c4} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} .$$

It is often useful to use only the first three elements of **q**$_{err}$ when performing attitude control. A quaternion error vector **q**$_e$ can thus be defined as

$$\mathbf{q}_e = \begin{bmatrix} q_{err,1} & q_{err,2} & q_{err,3} \end{bmatrix}^T , \tag{1.3.22}$$

which means that **q**$_{err}$ can also be expressed as

$$\mathbf{q}_{err} = \begin{bmatrix} \mathbf{q}_e \\ q_{err,4} \end{bmatrix} . \tag{1.3.23}$$

1.3.4 Satellite Dynamics and Kinematics

The **dynamics** of any given satellite can be described by the Euler dynamic equation. This equation states that the satellite body's moment of inertia (MoI) matrix **J** multiplied by the satellite body's inertial-referenced angular acceleration $\dot{\boldsymbol{\omega}}_B^I$ must be equal to the sum of all the torques (both internal and external) affecting the satellite. These torques can be divided into two groups: control torques and disturbance torques.

The only control torques considered for this project were magnetic control torque **N**$_m$ and wheel control torque **N**$_w$ (which also implies a wheel angular momentum **h**$_w$). This project focused on only one internal disturbance torque, namely gyroscopic coupling torque **N**$_{gyro}$. Three external disturbance torques were considered, namely gravity gradient torque **N**$_{gg}$, aerodynamic torque **N**$_{aero}$, and wheel imbalance torque **N**$_{rw}$. If **N**$_c$ and **N**$_d$ are respectively the sum of the control torques and the disturbance torques, then the Euler dynamic equation can be expressed as

$$\mathbf{J}\dot{\boldsymbol{\omega}}_B^I = \mathbf{N}_c + \mathbf{N}_d , \tag{1.3.24}$$

$$\text{where} \quad \mathbf{N}_c = \mathbf{N}_m - \mathbf{N}_w ,$$

$$\mathbf{N}_d \approx \mathbf{N}_{gg} + \mathbf{N}_{aero} + \mathbf{N}_{rw} - \mathbf{N}_{gyro} ,$$

$$\text{and} \quad \mathbf{N}_{gyro} = \boldsymbol{\omega}_B^I \times (\mathbf{J}\boldsymbol{\omega}_B^I + \mathbf{h}_w) .$$

The **kinematics** (i.e. how the attitude changes) of the satellite is represented by $\dot{\mathbf{q}}$. If ω_{xo}, ω_{yo}, and ω_{zo} are the components of the orbit-referenced angular rate vector $\boldsymbol{\omega}_B^O$, then $\dot{\mathbf{q}}$ can be expressed as

$$
\dot{\mathbf{q}} \;=\; \frac{1}{2}
\begin{bmatrix}
0 & \omega_{zo} & -\omega_{yo} & \omega_{xo} \\
-\omega_{zo} & 0 & \omega_{xo} & \omega_{yo} \\
\omega_{yo} & -\omega_{xo} & 0 & \omega_{zo} \\
-\omega_{xo} & -\omega_{yo} & -\omega_{zo} & 0
\end{bmatrix}
\mathbf{q} \; .
\tag{1.3.25}
$$

$\boldsymbol{\omega}_B^O$ is related to $\boldsymbol{\omega}_B^I$ through

$$
\boldsymbol{\omega}_B^O \;=\; \boldsymbol{\omega}_B^I - \boldsymbol{\omega}_O^I \; ,
\tag{1.3.26}
$$

where $\boldsymbol{\omega}_O^I$ is the angular rate of the ORC frame about the IRC frame with respect to the satellite body. Figure 1.6 implies that the ORC frame rotates about the Y-axis of the IRC frame once per orbit. The rate of this rotation is equal to $-\omega_o$, where ω_o is the angular rate of the satellite about the centre of the earth (i.e. the orbit rate). $\boldsymbol{\omega}_B^O$ can thus be expressed as

$$
\boldsymbol{\omega}_B^O \;=\; \boldsymbol{\omega}_B^I - \mathbf{A}
\begin{bmatrix}
0 \\
-\omega_o \\
0
\end{bmatrix} \; .
\tag{1.3.27}
$$

The DCM in Equation 1.3.27 is necessary to convert $\boldsymbol{\omega}_O^I$ from ORC to SBC.

1.3.5 Small Angle Approximation

The small angle approximation is a helpful tool when trying to solve complex ADCS problems. This approximation implies that if two arbitrary angles α and β are close to zero, then $\sin(\alpha) \approx \alpha$, $\sin(\beta) \approx \beta$, $\cos(\alpha) \approx \cos(\beta) \approx 1$, and $\alpha\beta \approx 0$. Under this approximation the direction cosine matrix $\mathbf{A}$ describing the satellite's attitude in the Euler-213 rotation sequence (see Equation 1.3.10) becomes

$$
\begin{aligned}
\mathbf{A} &\approx
\begin{bmatrix}
(1)(1) + (\theta)(\phi)(\psi) & (1)(\psi) & -(\theta)(1) + (1)(\phi)(\psi) \\
-(1)(\psi) + (\theta)(\phi)(1) & (1)(1) & (\theta)(\psi) + (1)(\phi)(1) \\
(\theta)(1) & -(\phi) & (1)(1)
\end{bmatrix} \\[2mm]
&\approx
\begin{bmatrix}
1 & \psi & -\theta \\
-\psi & 1 & \phi \\
\theta & -\phi & 1
\end{bmatrix} \; .
\end{aligned}
\tag{1.3.28}
$$

1.3.6   Orbit Parameters

The orbit of a satellite can be described by a minimum of five parameters, which are (1) semimajor axis a, (2) eccentricity e, (3) inclination i, (4) right ascension of the ascending node (RAAN) Ω, and (5) argument of perigee (AP) ω. The shape of the orbit is determined by a and e, whereas the orientation of the orbit relative to the EIC frame is dependent on i, Ω, and ω.

The inclination i is defined as the angle between the orbit plane and the equatorial plane. Ω is the angle between the Vernal Equinox Υ and the ascending node (i.e. the point where the satellite crosses the equatorial plane from South to North). ω is furthermore defined as the angle between the nodal line (which is the line between the ascending and descending nodes) and the orbit perigee. Figure 1.8 illustrates the above-mentioned orbit orientation.
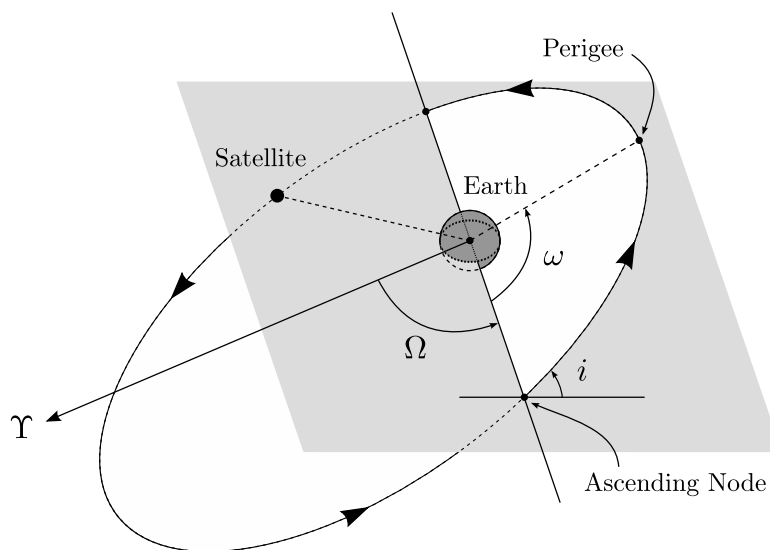
Figure 1.8 – The orientation of a satellite's orbit

The orbit of a satellite at a certain time instance can be described by a two-line element (TLE) set. Each set of two lines (with 69 characters each) is generated by the North American Aerospace Defence Command (NORAD). A TLE contains the satellite's NORAD identification number, a time-stamp (or epoch), the orbit parameters (e, i, Ω, and ω), a drag term B^*, the satellite's mean anomaly M, and its mean motion n (in revolutions per day). An orbit propagator can use the TLE data to calculate the satellite's position and velocity at any given time.

1.3.7   Forward Motion Compensation

An imaging payload on-board an EO satellite has a certain integration time during which data is captured by the image sensor. The higher the integration time, the more data can be captured, which implies that the image quality is heavily dependent on the integration time. The high ground speed of

low Earth orbit (LEO) satellites thus poses challenges when striving towards high quality images. The effective ground speed of the image sensor boresight can however be lowered by rotating the satellite to compensate for its forward motion, thus resulting in a higher image quality. This manoeuvre is known as forward motion compensation (FMC).

The basic idea behind FMC is to rotate the satellite about Y_O at a rate of $-\omega_{fmc}$ for a period of T_{fmc}. This rotation will cause the ground speed of the satellite to be lowered by a factor n_{fmc}. If the length of the image is L_{image} and the angle that the satellite will travel through while imaging is α, then the effective angle of the image is $\frac{\alpha}{n_{fmc}}$. The satellite will however need to be at a certain pitch angle θ_{fmc} when the image sensor starts collecting data. θ_{fmc} is a function of L_{image}, n_{fmc} and h (the satellite's altitude). An illustration of the FMC imaging manoeuvre can be seen in Figure 1.9.
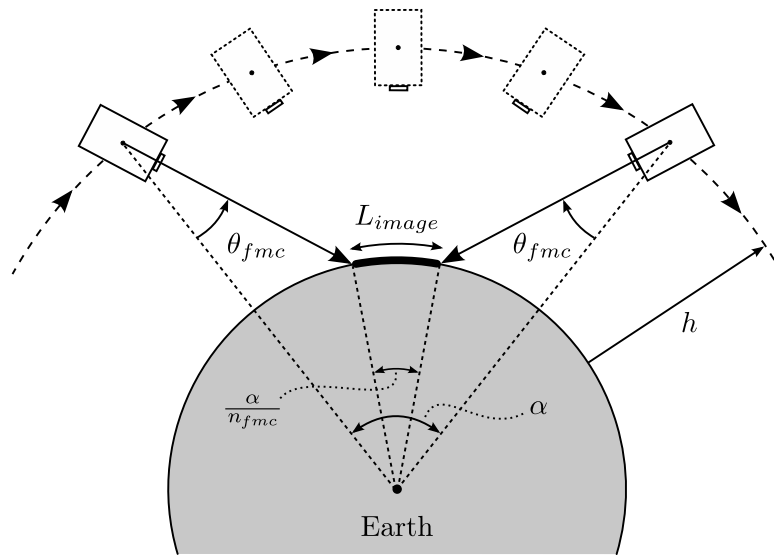
Figure 1.9 – An illustration of the FMC imaging manoeuvre

The FMC parameters shown in Figure 1.9 can easily be calculated if h, ω_o, L_{image}, and n_{fmc} are known. The satellite's ground speed v_g is firstly calculated using the Earth's radius R_E through

$$v_g = \omega_o R_E \,, \tag{1.3.29}$$

after which ω_{fmc} can be determined via

$$\omega_{fmc} = \frac{v_g}{h}\left(1 - \frac{1}{n_{fmc}}\right) \,. \tag{1.3.30}$$

The FMC duration T_{fmc} can furthermore be calculated as

$$T_{fmc} = \frac{n_{fmc} L_{image}}{v_g} \,. \tag{1.3.31}$$

Figure 1.9 also implies that the satellite will pitch through an angle of $2\theta_{fmc}$ during imaging. The total pitch rotation angle is also equal to the pitch rate ω_{fmc} multiplied by the duration T_{fmc}, which means that θ_{fmc} can be expressed as

$$\theta_{fmc} \quad = \quad \frac{1}{2}\omega_{fmc}T_{fmc} \; . \tag{1.3.32}$$

Although the angles α and $\frac{\alpha}{n_{fmc}}$ are not used or calculated by the on-board computer of the satellite, they still provide insight into the working of the FMC imaging manoeuvre. The effective angle of the image, or $\frac{\alpha}{n_{fmc}}$, can be calculated as

$$\frac{\alpha}{n_{fmc}} \quad = \quad \frac{R_E}{L_{image}} \; , \tag{1.3.33}$$

which means that the satellite's orbit angle α during imaging can be expressed as

$$\alpha \quad = \quad \frac{n_{fmc}R_E}{L_{image}} \; . \tag{1.3.34}$$

Two special cases for FMC imaging are $n_{fmc} = 1$ and $n_{fmc} = \infty$. If $n_{fmc} = 1$, ω_{fmc} will be zero, which implies push-broom imaging. If $n_{fmc} = \infty$, the effective angle of the image will be zero, which implies target tracking [3].

1.4   Document Outline

This document consists of eight chapters. The subsections below will summarise the main focus points of each chapter.

Chapter 1: Introduction

The first (and current) chapter is dedicated to introducing the purpose of and various concepts used in the project at hand. An investigation into the current trends regarding small satellites is also discussed in this chapter.

Chapter 2: Simulation Environment

The ADCS simulation environment that was created using MATLAB and Simulink will be discussed in this chapter. The various space environment models that were implemented in the simulation environment will be mentioned, after which three disturbance models will be derived. The process of simulating the satellite's dynamics and kinematics will also be described in this chapter.

Chapter 3: Initial ADCS Analysis

This chapter will focus on the initial ADCS analysis of the satellite on which this project is based. A given set of satellite requirements and specifications will be translated into minimum sensor and actuator requirements in the presence of worst case disturbances.

Chapter 4: An Application Specific Simulation Setup

The hardware chosen for the 20 kg satellite was modelled in the simulation environment and the derivation of these models will be shown in this chapter. Emphasis will furthermore be placed on sensor noise and placement. The practical example that was used to derive the reaction wheel model will also be discussed.

Chapter 5: Attitude Determination

Various attitude determination methods will be derived in this chapter. The performance of each estimator was determined through simulation and the results thereof will be shown. Notes on the practical implementation of the estimators will also be given.

Chapter 6: Attitude Control

This chapter will be divided into two parts: magnetic control and wheel control. Numerous attitude controllers will be derived and discussed and the simulation results confirming the correct functioning of each controller will also be shown.

Chapter 7: An Application Specific ADCS

An ADCS strategy for the 20 kg Earth observation satellite on which this project is based will be suggested in this chapter. The state machine driving the ADCS and the five chosen ADCS modes will furthermore be commented on. Simulation results of each of the modes will be given, including those of FMC imaging and target tracking manoeuvres. The performance of the satellite with respect to the given set of requirements will lastly be analysed.

Chapter 8: Conclusions and Recommendations

The final chapter of this document will be dedicated to summarising the conclusions that were drawn from the work done during the course of this project. Recommendations and possible improvements to the project will also be given.

1.5   Summary

This introductory chapter was firstly aimed at introducing the purpose of and need for the research on which this document is based by means of a problem statement. The findings of an investigation into the current trends regarding small satellites, their uses, and the ADCS hardware they employ was also discussed. Furthermore, a few of the fundamental ADCS concepts that were utilised during the course of this project were discussed in detail. The outline of this document was briefly described in conclusion of this chapter.

Chapter 2

Simulation Environment

2.1   Overview

This chapter will elaborate on the ADCS simulation environment that was developed (using MATLAB and Simulink) as part of this project. The modelling of the satellite's environment will firstly be covered, after which a few sources of disturbance will be discussed. The process of simulating the satellite's dynamics and kinematics will lastly be summarised.

The satellite environment models that were implemented include an orbit propagation tool, a geomagnetic field model and a sun position model. Output examples of each of these environment models will be shown in this chapter. Only three sources of disturbance were chosen to be modelled for the project at hand, namely gravity gradient, aerodynamic disturbance, and wheel imbalance. This chapter will also briefly discuss the fundamental theory behind each of the above-mentioned disturbance sources. Lastly, the implementation of the satellite dynamics and kinematics model will be covered.

It should be noted that some of the investigated models were not only developed for simulating the satellite's true dynamics, kinematics, and environment, but also for satellite dynamics and kinematics estimation.

2.2   Space Environment Models

Models of the space environment are required to simulate the satellite in its orbit and to propagate its dynamics and kinematics. The environment of an Earth-orbiting satellite furthermore needs to be modelled accurately for ADCS purposes, since measured vectors are compared against modelled vectors in the process of attitude determination. The elements of the space environment that were modelled during the course of this project will be discussed in this section.

2.2.1 Orbit Propagation

The orbit of a satellite at a certain time instance can be described by a TLE set. The ADCS simulation environment uses the given TLE as a starting point to propagate the satellite's orbit. An ADCS requires knowledge of the current position $\mathbf{r}_{sat}$ and velocity $\mathbf{v}_{sat}$ (both in ECI coordinates) of the satellite for accurate determination and control. $\mathbf{r}_{sat}$ and $\mathbf{v}_{sat}$ can be generated using a simplified general perturbations (SGP) model.

Although the first of the SGP models was developed in the 1960s, a widely-used refined version known as SGP4 was published in 1980 in SPACETRACK REPORT NO.3 (STR#3) [4]. The numerous improvements that were made to the SGP4 over the years were summarised by Vallado et al. in 2006 [5]. In the project at hand, STR#3 was used to create an initial orbit propagator, after which the suggested improvements were implemented. The improved SGP4 uses the World Geodetic System 72 constants and serves as the orbit propagator in the ADCS simulation environment. Figure 2.1 shows three different orbits propagated by the developed SGP4.
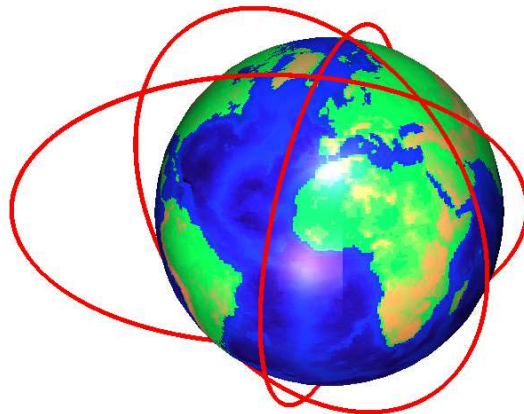
Figure 2.1 – Examples of satellite orbits propagated by the SGP4 in the ADCS simulation environment

The inputs to the SGP4 are elapsed time Δt and TLE data. A simple function was thus created to extract data from a TLE based on a given NORAD identification number. The SGP4 then propagates the orbit of the satellite from t_0, the TLE epoch time, to $t = t_0 + \Delta t$. The outputs of the SGP4 include the satellite's EIC position and velocity vectors $\mathbf{r}_{sat}$ and $\mathbf{v}_{sat}$, the orbit radius r_s, and the orbit rate ω_o. $\mathbf{r}_{sat}$ and $\mathbf{v}_{sat}$ can also be used to calculate $\mathbf{A}_{EIC}^{ORC}$ using Equation 1.3.5.

The current time t is furthermore used to determine the current Julian Date (JD), which in turn is then used to determine the Greenwich hour angle α_G. $\mathbf{r}_{sat}$, t, α_G, and the current JD are inputs to a function that determines the current satellite latitude φ, longitude λ, and altitude h, all of which are used by the other models in the ADCS simulation environment.

2.2.2 Geomagnetic Field

The Earth's magnetic field plays an important role in the dynamics of a satellite. A magnetic dipole moment generated by the satellite's attitude control system (or a residual magnetic dipole moment) will react with the geomagnetic field to produce a magnetic torque. The measured geomagnetic field is often also compared to the modelled geomagnetic field in the process of attitude determination. Modelling the geomagnetic field is thus a fundamental element of both the satellite's ADCS and the simulation environment.

Although the geomagnetic field can be approximated by a simple dipole model, a more sophisticated model is required for ADCS purposes. The Earth's magnetic field also varies slowly with time, which means that a time-dependent model is necessary for increased ADCS accuracy. The International Geomagnetic Reference Field (IGRF) model is a geomagnetic field model released by the International Association of Geomagnetism and Aeronomy (IAGA). The model provides a numerical representation of the geomagnetic field and is currently in its 11th generation, which was released in December 2009 [6].

The geomagnetic field strength $\mathbf{B}$ is calculated as the negative gradient of the scalar potential function V [2], thus

$$\mathbf{B} \;=\; -\nabla V \; . \tag{2.2.1}$$

V is a function of the orbit radius r_s, coelevation θ (which is 90° minus latitude φ), and longitude λ. The scalar potential function can furthermore be expressed as a series expansion of spherical harmonics, i.e.

$$V(r_s, \theta, \lambda) \;=\; R_E \sum_{n=1}^{k} \left(\frac{R_E}{r_s} \right)^{n+1} \sum_{m=0}^{n} \Big(g_n^m \cos\left(m\lambda\right) + h_n^m \sin\left(m\lambda\right) \Big) P_n^m(\theta) \; , \tag{2.2.2}$$

where k is the order of the expansion, g_n^m and h_n^m are coefficients defined by the IGRF and $P_n^m(\theta)$ is the Legendre function with respect to m, n, and θ [2]. $\mathbf{A}_{EIC}^{ORC}$ is lastly used to transform the IGRF-generated geomagnetic field from EIC to ORC.

Although IAGA provides higher order coefficients, only a 10th order IGRF model (with the 2009 coefficients propagated to 2014) is implemented in the ADCS simulation environment. Figure 2.2 shows a typical example of $\mathbf{B}_o$, the geomagnetic field in ORC, for a 500 km near-polar orbit. As expected for a near-polar orbit, the Y-component of $\mathbf{B}_o$ in Figure 2.2 remains small throughout the orbit. The maximum and minimum B_{ox} values are reached near the equatorial regions, whereas B_{oz} reaches its maximum and minimum values near the poles.

Although the satellite will model (and thus have access to) $\mathbf{B}_o$ based on the IGRF, the true orbit-referenced geomagnetic field will be subject to noise [2] which is of a low frequency nature. Noise with a maximum amplitude of 1 μT was thus added to model the above-mentioned.
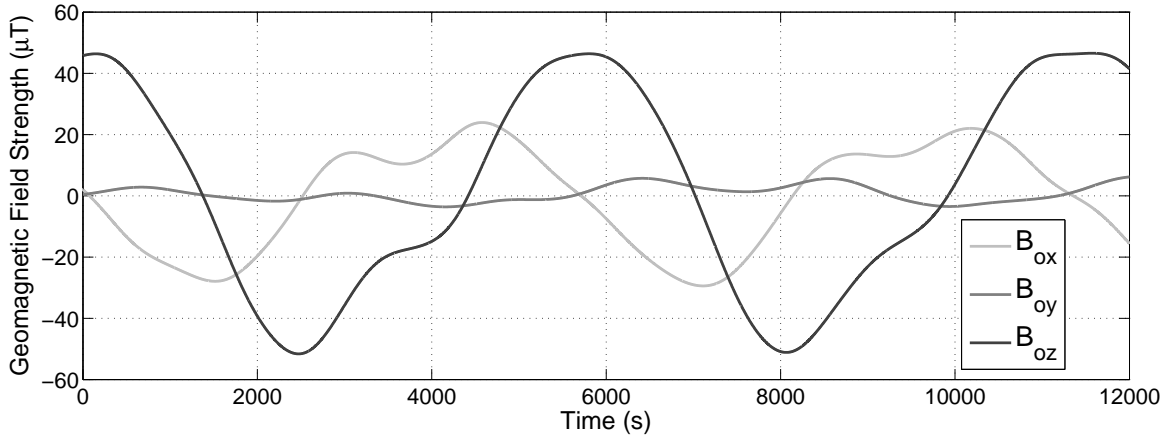
Figure 2.2 – The geomagnetic field vector generated by the 10$^{\text{th}}$ order IGRF model

2.2.3   Sun Position

Sun sensors are widely used on satellites to assist with the process of illuminating the solar panels or to estimate attitude. The on-board computer uses both the measured unit sun vector $\mathbf{S}_b$ (in SBC) and the modelled unit sun vector $\mathbf{S}_o$ (in ORC) when estimating the satellite's attitude and angular rates and when performing sun tracking control. A sun model was therefore included in the ADCS simulation environment.

The chosen model [7] provides an ECI frame sun vector $\mathbf{r}_{sun}$, which is then used to calculate $\mathbf{S}_o$. The first step of the model is to calculate T_{JC}, the amount of Julian centuries (i.e. 365.25 days) between the current Julian Date J_t and the J2000 epoch (i.e. the JD of 12:00, 1 January 2000, which is 2451545). T_{JC} can thus be expressed as

$$T_{JC} \;=\; \frac{J_t - 2451545}{36525} \; . \tag{2.2.3}$$

A series of equations are then used to calculate the sun's mean longitude $\lambda_{M_\odot}$ and mean anomaly $M_\odot$, the ecliptic longitude λ_e and obliquity ϵ, and the sun position magnitude $r_\odot$. The ECI sun vector is approximated in astronomical units (AU, the mean distance from the Earth to the sun) as

$$\mathbf{r}_{sun} \;=\; r_\odot \begin{bmatrix} \cos\left(\lambda_e\right) \\ \cos\left(\epsilon\right)\sin\left(\lambda_e\right) \\ \sin\left(\epsilon\right)\sin\left(\lambda_e\right) \end{bmatrix} \; , \tag{2.2.4}$$

$$\text{where} \quad r_\odot \;=\; 1.000\,140\,612 - 0.016\,708\,617\cos\left(M_\odot\right) - 0.000\,139\,589\cos\left(2M_\odot\right) \; ,$$

$$M_\odot \;=\; 357.527\,723\,300° + 35\,999.050\,340\,T_{JC} \; ,$$

$$\lambda_e \;=\; \lambda_{M_\odot} + 1.914\,666\,471\sin\left(M_\odot\right) + 0.019\,994\,643\sin\left(2M_\odot\right) \; ,$$

$$\lambda_{M_\odot} \;=\; 280.460\,618\,400° + 36\,000.770\,053\,610\,T_{JC} \; ,$$

$$\text{and} \quad \epsilon \;=\; 23.439\,291° - 0.013\,004\,200\,T_{JC} \; .$$

$\mathbf{r}_{sun}$ is then converted to a vector in km (with 1 AU = 149 597 871 km [8]), after which the EIC vector from the satellite to the sun ($\mathbf{S}_{EIC}$) is calculated using

$$\mathbf{S}_{EIC} = \mathbf{r}_{sun} - \mathbf{r}_{sat} . \tag{2.2.5}$$

$\mathbf{S}_{EIC}$ is then normalised and transformed to $\mathbf{S}_o$ using $\mathbf{A}_{EIC}^{ORC}$.

The position of the sun is also required to determine whether or not the satellite is in eclipse. The ADCS simulation environment uses simple geometry (involving the use of $\mathbf{r}_{sun}$ (in km), $\mathbf{r}_{sat}$, a vector dot-product operation, and the cosine rule) to determine if the sun is in the line of sight of the satellite. For the project at hand it was defined that the satellite is in eclipse if any part of the sun is obstructed by the Earth. Figure 2.3 illustrates the eclipse definition by showing the above-mentioned geometry.
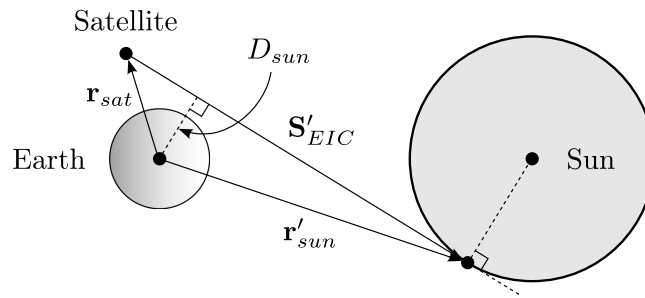
Figure 2.3 – The satellite eclipse geometry

$\mathbf{r}'_{sun}$ in Figure 2.3 is the EIC vector from the centre of the Earth to the lowest point on the sun in the satellite's field of view (FOV). Similarly, $\mathbf{S}'_{EIC}$ is the EIC vector from the satellite to the above-mentioned point on the sun. D_{sun} is the distance from the centre of the Earth to the vector $\mathbf{S}'_{EIC}$. From Figure 2.3 it can be derived that if $D_{sun} < R_E$, the satellite will be in eclipse.

Figure 2.4 shows a typical example of $\mathbf{S}_o$ for a 500 km sun-synchronous orbit (SSO) with eclipse applied to the vector. As expected for this type of orbit, both the Y-component of the unit sun vector and the eclipse duration in Figure 2.4 remain constant.

2.3 Disturbance Models

Any Earth-orbiting satellite is affected by a number of disturbances. These disturbances create torques that affect the satellite's dynamics. Three major sources of disturbance were modelled in the ADCS simulation environment, namely gravity gradient, aerodynamic disturbance, and wheel imbalance. Other sources of disturbance that were not modelled include solar radiation pressure (SRP), residual magnetic dipole moment, thruster misalignment, fuel slosh, and motor torque ripple.
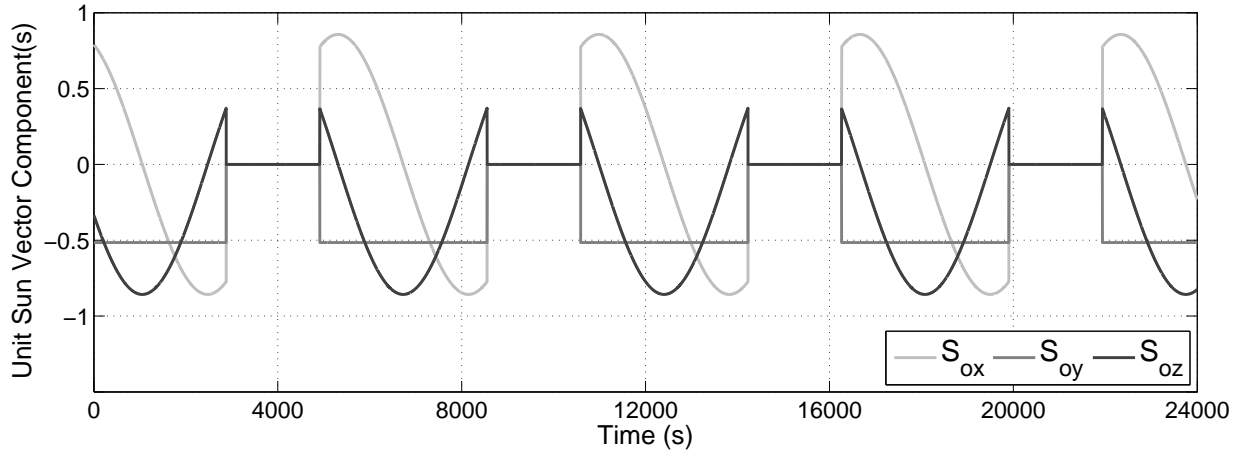
Figure 2.4 – The unit sun vector generated by the sun position model

2.3.1 Gravity Gradient

The gravity gradient disturbance torque ($\mathbf{N}_{gg}$) is a result of both gravitational force and centrifugal force. The part of the satellite body that is closest to the Earth will experience a greater gravitational force (i.e. towards the Earth), whereas the furthest part of the satellite will experience a greater centrifugal force (i.e. away from the Earth). $\mathbf{N}_{gg}$ can also easily be calculated via

$$\mathbf{N}_{gg} \quad = \quad 3\,\omega_o^2\left(\mathbf{z}_o^B \times \mathbf{J}\mathbf{z}_o^B\right)\,, \tag{2.3.1}$$

where $\mathbf{z}_o^B$ is the orbit nadir vector in SBC, thus

$$\mathbf{z}_o^B \quad = \quad \mathbf{A}\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T\,. \tag{2.3.2}$$

If it is assumed that the satellite's products of inertia are negligible compared to the moments of inertia I_{xx}, I_{yy}, and I_{zz} (i.e. $\mathbf{J} = \mathbf{I}_{3\times3}\begin{bmatrix} I_{xx} & I_{yy} & I_{zz} \end{bmatrix}^T$), then $\mathbf{N}_{gg}$ can be expressed in terms of the elements of the DCM $\mathbf{A}$ as

$$\mathbf{N}_{gg} \quad = \quad \begin{bmatrix} k_{gx}A_{23}A_{33} \\ k_{gy}A_{13}A_{33} \\ k_{gz}A_{13}A_{23} \end{bmatrix}\,, \tag{2.3.3}$$

$$\text{where} \quad k_{gx} \quad = \quad 3\omega_o^2(I_{zz} - I_{yy})\,,$$

$$k_{gy} \quad = \quad 3\omega_o^2(I_{xx} - I_{zz})\,,$$

$$\text{and} \quad k_{gz} \quad = \quad 3\omega_o^2(I_{yy} - I_{xx})\,.$$

$\mathbf{N}_{gg}$ is unique in the sense that it is the only disturbance torque that can be modelled accurately on-board a satellite and it can thus be used by the ADCS estimators.

2.3.2   Aerodynamic Disturbance

The atmospheric density in a LEO is high enough to create noticeable disturbance torques [9]. Several factors influence the magnitude and direction of the aerodynamic disturbance torque vector $\mathbf{N}_{aero}$, namely atmospheric density ρ, surface area A, the Earth's rotation rate ω_E, and the centre of mass (CoM) to centre of pressure (CoP) offset vector $\mathbf{r}_p$. The aerodynamic disturbance model determines $\mathbf{N}_{aero}$ by calculating the sum of the disturbance torques for each individual segment of the satellite [10]. For a satellite with n segments, $\mathbf{N}_{aero}$ is calculated as

$$
\mathbf{N}_{aero} = \sum_{i=1}^{n} \left(\rho \|\mathbf{v}_A^B\|^2 A_i H\{\cos(\alpha_i)\} \cos(\alpha_i) \Big(\sigma_t (\mathbf{r}_{pi} \times \overline{\mathbf{v}}_A^B) + \right.
$$
$$
\left. [\sigma_n S + (2 - \sigma_n - \sigma_t)\cos(\alpha_i)](\mathbf{r}_{pi} \times \overline{\mathbf{n}}_i) \Big) \right) , \tag{2.3.4}
$$

where $\mathbf{v}_A^B$ is the atmospheric velocity in SBC with the unit vector $\overline{\mathbf{v}}_A^B$, A_i is the surface area of segment i, $H\{...\}$ is the Heaviside function, α_i is the incidence angle of $\mathbf{v}_A^B$ on segment i, σ_t is the tangential accommodation coefficient, σ_n is the normal accommodation coefficient, S is the ratio of molecular exit velocity to $\mathbf{v}_A^B$, $\mathbf{r}_{pi}$ is segment i's CoM to CoP offset vector, and $\overline{\mathbf{n}}_i$ is the unit inward normal vector of segment i.

A static exponential atmospheric density model [7] is used in the ADCS simulation environment. This model calculates ρ as

$$
\rho = \rho_0 e^{-\frac{h - h_0}{H}} , \tag{2.3.5}
$$

where ρ_0 is the reference density, h is the satellite's current altitude, h_0 is the reference altitude, and H is the scale height. It is furthermore assumed that the the atmospheric density during eclipse is $\frac{1}{2}\rho$ [11].

The atmospheric velocity vector in EIC ($\mathbf{v}_A^{EIC}$) with respect to the satellite is given by

$$
\mathbf{v}_A^{EIC} = \begin{bmatrix} 0 \\ 0 \\ \omega_E \end{bmatrix} \times \mathbf{r}_{sat} - \mathbf{v}_{sat} . \tag{2.3.6}
$$

The first term in Equation 2.3.6 represents the effect of the atmosphere rotating with the Earth, whereas the second term represents the "collision" of the satellite with the atmosphere as a result of the satellite's velocity vector. $\mathbf{A}_{EIC}^{ORC}$ transforms $\mathbf{v}_A^{EIC}$ to ORC, after which the DCM is used to determine $\mathbf{v}_A^B$.

The surface area A_i is calculated from the satellite's dimensions and $\cos(\alpha_i)$ is calculated as the dot-product between $\overline{\mathbf{n}}_i$ and $\overline{\mathbf{v}}_A^B$. The Heaviside function, i.e.

$$
H\{x\} = 0 \quad \text{if } x < 0 ,
$$
$$
H\{x\} = 1 \quad \text{if } x \geq 0 ,
$$

is implemented in Equation 2.3.4 to ensure that only segments with an incidence angle lower than 90° will contribute towards $\mathbf{N}_{aero}$. σ_t and σ_n from Equation 2.3.4 were assumed to be 0.8, and S was set to 0.05 [11].

The accuracy of ρ, σ_t, σ_n, and S are not crucially important in the simulation environment. The attitude estimators will not have knowledge of $\mathbf{N}_{aero}$, which implies that the goal of the aerodynamic disturbance in the simulation environment is just to create some form of disturbance torque. The ADCS's ability to reject the disturbance torque will shed some light on the performance of the ADCS.

2.3.3   Wheel Imbalance

A momentum wheel or a reaction wheel (RW) used on a satellite is subject to various disturbances, for example flywheel imbalance, motor torque ripple and motor driver noise [12]. Only flywheel imbalance was chosen to be modelled in the ADCS simulation environment, since it is "often considered to be the most significant source" of wheel-induced disturbance torque [13]. Although flywheels are often manufactured with precision (i.e. low tolerances), the mass of a flywheel will almost never be distributed evenly. The uneven mass distribution (or mass imbalance) of a flywheel can divided into a static imbalance and a dynamic imbalance.

Static imbalance is the CoM offset from the axis of rotation and is modelled as a small mass m on the edge of a flywheel with radius r. The rotating mass generates a radial force in the plane perpendicular to the axis of rotation. For example, if the Z-wheel static imbalance is $U_s = mr$, the rotation rate is ω, t is time, and ϕ_s is an arbitrary phase, then the rotating radial force $\mathbf{F}_{zs}$ in SBC can be expressed as

$$\mathbf{F}_{zs} \;=\; U_s\omega^2 \begin{bmatrix} \sin\left(\omega t + \phi_s\right) \\ \cos\left(\omega t + \phi_s\right) \\ 0 \end{bmatrix} . \tag{2.3.7}$$

The disturbance torque generated as a result of $\mathbf{F}_{zs}$ is a function of the location of the wheel. For example, if the position vector of the Z-wheel with respect to the satellite's CoM is $\mathbf{w}_z$, then the static imbalance disturbance torque $\mathbf{N}_{zs}$ can be expressed as

$$\mathbf{N}_{zs} \;=\; \mathbf{w}_z \times \mathbf{F}_{zs} . \tag{2.3.8}$$

Dynamic imbalance is the misalignment of the principal inertia with respect to the axis of rotation, thus representing the flywheel's cross product inertia [12]. The dynamic imbalance is modelled by two equal masses m separated by 180° and axially spaced by d. The two rotating masses result in two opposite radial forces in the plane perpendicular to the axis of rotation. For the Z-wheel, for example, the magnitude F_{zd} of each force is equal to

$$F_{zd} \;=\; mr\omega^2 . \tag{2.3.9}$$

The resulting Z-wheel dynamic imbalance torque $\mathbf{N}_{zd}$ is calculated as

$$\mathbf{N}_{zd} \quad = \quad U_d \omega^2 \begin{bmatrix} \sin\left(\omega t + \phi_d\right) \\ \cos\left(\omega t + \phi_d\right) \\ 0 \end{bmatrix} \, , \tag{2.3.10}$$

where $U_d = mrd$ is the dynamic imbalance of the wheel and ϕ_d is once again an arbitrary phase. Figure 2.5 illustrates the above-mentioned wheel imbalance models.

(a) Static imbalance   (b) Dynamic imbalance
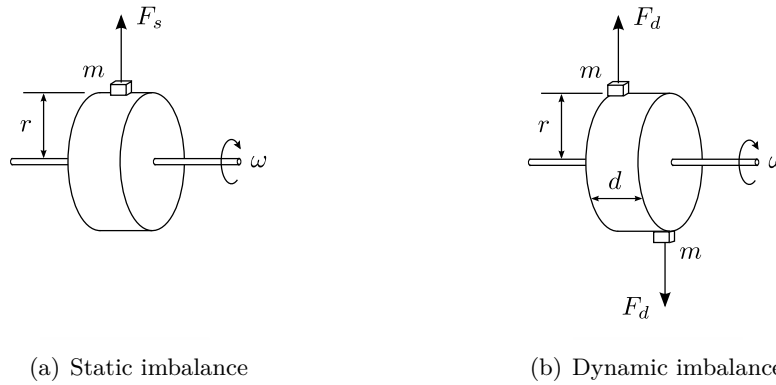
Figure 2.5 – An illustration of the wheel imbalance disturbance torque

2.4   Satellite Dynamics and Kinematics

The most important model in the ADCS simulation environment is that of the satellite's dynamics and kinematics. Even though the angular rates and attitude of the satellite are estimated by the ADCS in practice, an ideal (or true) model thereof is required to determine the performance of the ADCS during simulation.

The dynamic model of the satellite (see Equation 1.3.24) is implemented in the simulation environment using continuous Simulink blocks. This model is affected by all the disturbance models, as well as by the true actuator outputs. The output of Equation 1.3.24, $\dot{\boldsymbol{\omega}}_B^I$, is integrated in the continuous domain using Simulink's 2$^{\text{nd}}$ order (or Heun) solver, which produces the true inertial-referenced angular rate vector $\boldsymbol{\omega}_B^I$.

The kinematic model of the satellite is similarly implemented by integrating Equation 1.3.25 in the continuous domain to compute $\mathbf{q}$. The orbit-referenced angular rate vector $\boldsymbol{\omega}_B^O$ is also calculated (using Equation 1.3.27) as part of this process. Figure 2.6 shows $\boldsymbol{\omega}_B^I$, $\boldsymbol{\omega}_B^O$, and the Euler-213 angles (as calculated from $\mathbf{q}$) for an example case where the initial rates and attitude was set to $\boldsymbol{\omega}_B^I = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T {}^\circ/\text{s}$ and $\mathbf{q} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$ for a satellite with $\begin{bmatrix} I_{xx} & I_{yy} & I_{zz} \end{bmatrix} = \begin{bmatrix} 3 & 6 & 1 \end{bmatrix}$ kgm^2.

(a) Inertial-referenced body rates   (b) Orbit-referenced body rates   (c) Euler angles
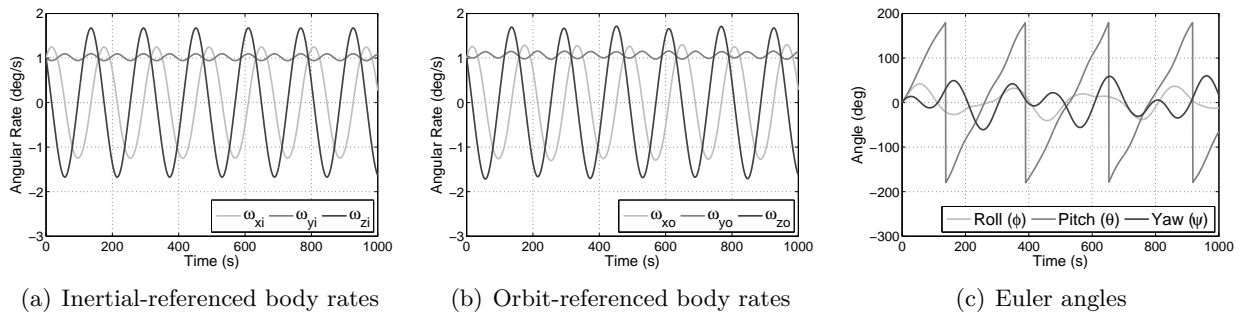
Figure 2.6 – Example outputs of the satellite dynamics and kinematics model

2.5   Summary

The ADCS simulation environment that was developed during the course of this project was discussed in detail in this chapter. The space environment was firstly described, after which some of the various sources of disturbance were covered in depth. The model of the satellite's dynamics and kinematics was also discussed.

The SGP4 was chosen as the orbit propagation tool for this project. A 10th order IGRF model and a sun position model was also implemented to produce reference geomagnetic field and unit sun vectors in ORC, both of which are also used by the satellite's ADCS. Example outputs of the three above-mentioned space environment models were also shown.

Although numerous sources of disturbance exist, only three were modelled in the simulation environment, namely gravity gradient, aerodynamic disturbance, and wheel imbalance. The satellite's dynamics and kinematics model was implemented using Simulink's 2nd order continuous domain solver.

The simulation environment described in this chapter (of which a basic overview can be seen in Figure 2.7) served as the platform from which the various ADCS simulation results that will be shown in Chapters 5 to 7 were produced.
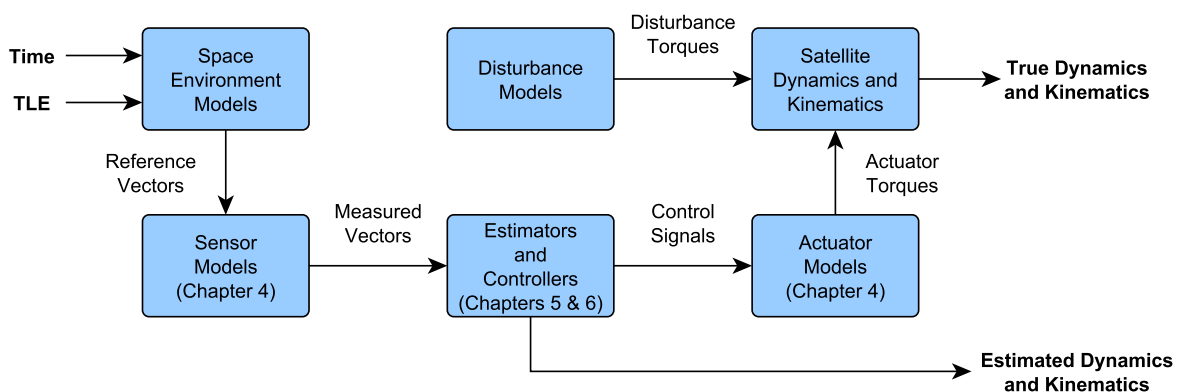
Figure 2.7 – An overview of the simulation environment

Chapter 3

Initial ADCS Analysis

3.1   Overview

The selection ADCS hardware for a satellite mission is heavily dependent on the mission type and the satellite specifications. For example, a space exploration satellite will have little use for an Earth horizon sensor, and an Earth observation satellite will require a high accuracy sensor for precision pointing during imaging.

This chapter will firstly discuss the specifications and requirements of the satellite on which this project is based, after which the given requirements will be translated into sensor and actuator requirements. A position and pointing budget will be constructed to assist with the high accuracy sensor selection process, whereas the minimum actuator specifications will be determined from the agility requirement and the maximum disturbance torques. An analysis on the worst case jitter as a result of the chosen actuators will be done in conclusion of this chapter.

3.2   Satellite Specifications

Before the ADCS hardware can be chosen, the satellite specifications must be investigated. The specifications influence the choice of hardware and in a number of different ways. Although the orbit type for Earth observation is generally sun-synchronous, the altitude and local time of the ascending node (LTAN) of the orbit will influence the placement of sensors on the satellite body. The pointing knowledge requirement will determine the high accuracy ADCS sensor, whereas the type and size of the actuators will be dependent on the pointing accuracy, agility, and stability requirements.

The following specifications are given for the EO satellite:

- Mass: 20 kg
- Dimensions: $L_x \times L_y \times L_z = 0.3$ m $\times$ 0.3 m $\times$ 0.4 m

- Moment of inertia (MoI) elements: $\begin{bmatrix} I_{xx} & I_{yy} & I_{zz} \end{bmatrix} = \begin{bmatrix} 0.4 & 0.45 & 0.3 \end{bmatrix}$ kgm^2

- Mission lifetime: 3 - 5 years

- Orbit: 500 km SSO

- Position knowledge: < 500 m

- Pointing accuracy: < 2 km

- Agility: 30° slew in 30 s

- Stability: $< 10\%$ image pixel smear

- Image ground sampling distance (GSD): 10 m

- Image swath width: 40 km

- Image FMC factor: 4

The stability specification is better understood from an ADCS point of view when expressed as an angular rate. The maximum allowable angular rate variation is the change in angular rate that would cause 10% pixel smear during the image exposure time t_{exp}. The GSD (i.e. one pixel) should thus be converted to an angular distance θ_{pixel} through

$$
\begin{aligned}
\theta_{pixel} \quad &= \quad \tan^{-1}\left(\frac{\text{GSD}}{h}\right) \\
&= \quad 4.125 \text{ arcsec .}
\end{aligned}
\tag{3.2.1}
$$

The exposure time t_{exp} of a single pixel (or a single row of pixels) is a function of the effective ground speed $v_{g,eff}$. For an FMC factor of $n_{fmc} = 4$, the effective ground speed during imaging is 1.7648 km/s, as calculated from the ground speed v_g and the satellite speed v_{sat} through

$$
v_{g,eff} \quad = \quad \frac{v_g}{n_{fmc}} \ ,
\tag{3.2.2}
$$

$$
\text{where} \quad v_g \quad = \quad v_{sat}\left(\frac{R_E}{R_E + h}\right)
\tag{3.2.3}
$$

$$
\text{and} \quad v_{sat} \quad \approx \quad \sqrt{\frac{\mu_E}{R_E + h}} \ .
\tag{3.2.4}
$$

μ_E in Equation 3.2.4 is the Earth's gravitational constant. The exposure time for one 10 m $\times$ 10 m ground pixel can thus be calculated as

$$
\begin{aligned}
t_{exp} \quad &= \quad \frac{\text{GSD}}{v_{g,eff}} \\
&= \quad 5.67 \text{ ms .}
\end{aligned}
\tag{3.2.5}
$$

For a maximum of 10% pixel smear, the allowable change in angular rate about the image sensor's boresight, or ω_{sz}, can be determined as

$$
\begin{aligned}
\omega_{sz} &\approx \frac{\tan^{-1}\left(\frac{0.1\times\text{GSD}}{\frac{1}{2}L_{swath}}\right)}{t_{exp}} \\
&= 1819 \text{ arcsec/s} ,
\end{aligned}
\tag{3.2.6}
$$

where L_{swath} is the image swath width of 40 km. The allowable change in angular rate ω_{sxy} across the image sensor's boresight can similarly be calculated via

$$
\begin{aligned}
\omega_{sxy} &= \frac{0.1 \times \theta_{pixel}}{t_{exp}} \\
&= 72.8 \text{ arcsec/s} .
\end{aligned}
\tag{3.2.7}
$$

The ADCS actuators must thus ensure stability better than 72.8 arcsec/s during imaging. For FMC, the last-mentioned implies that the angular rate about the orbit Y-axis may not differ from the desired rate ω_{fmc} by more than 72.8 arcsec/s. Figure 3.1 illustrates the underlying geometry of Equations 3.2.6 and 3.2.7. t_0 in Figure 3.1 indicates the start time of the image sensor integration.
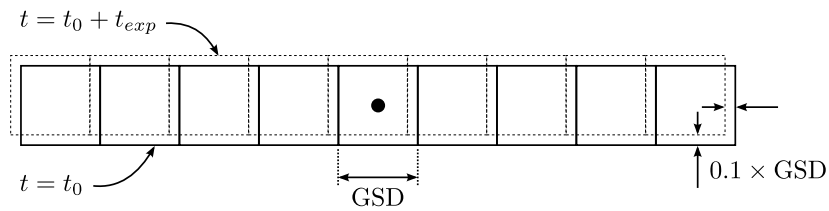
(a) Across-boresight pixel smear
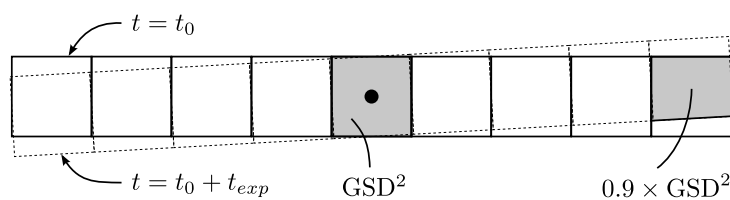
(b) Around-boresight pixel smear

Figure 3.1 – Pixel smear geometry during imaging

3.3   Position and Pointing Budget

Several factors contribute to the position knowledge error and pointing error of a satellite. The effect of some of these factors can be approximated or assumed, whereas others are dependent on the ADCS hardware. A position and pointing budget can be constructed to determine the required sensor accuracy and control accuracy of the ADCS.

The various factors affecting the position knowledge error should firstly be quantified. The total error, which includes the error from the most accurate sensor, must not exceed the specified maximum position knowledge error (which is 500 m for this project). Since the latter is known, the variable in the budget, i.e. the maximum sensor attitude knowledge error, can be determined. The result can be used to select an appropriate high accuracy ADCS sensor. Furthermore, the difference between the maximum pointing error and the maximum position knowledge error indicates the maximum control error.

The position and pointing budget for the 20 kg EO satellite can be seen in Table 3.1.

	3-σ error	**3-σ error (m)**
GPS position error	20 m	20
Software latency	15 ms	106
Structural alignment error	0.01°	87
Maximum attitude estimation error	**118 arcsec**	287
Maximum position knowledge error	206 arcsec	500
Maximum control error	**619 arcsec**	1500
Maximum pointing error	825 arcsec	2000

Table 3.1 – The position and pointing budget

Table 3.1 indicates that a high accuracy sensor with a 3-σ error of less than 118 arcsec will be required. Furthermore, fine-pointing actuators (for example reaction wheels) will need to be employed to reach the required control accuracy of less than 619 arcsec. The last-mentioned is also apparent from the satellite mission type (i.e. Earth observation), which requires precision pointing.

3.4   Agility

The agility specification from Section 3.2 will determine the minimum required angular momentum and torque capabilities of the reaction wheels used on the satellite. In the agility analysis it is assumed that the slew will be performed using the maximum reaction wheel torque $N_{ws,max}$ (i.e. a Bang-Bang control strategy). The maximum angular acceleration $\ddot{\theta}_{max}$ is also known to be directly proportional to $N_{ws,max}$. The maximum angular momentum $h_{ws,max}$ (and also the maximum angular rate $\dot{\theta}_{max}$)

for a Bang-Bang controller will be reached at $\frac{1}{2}t_f$, where t_f is the agility requirement's slew period (which is 30 s for this project). If θ_f is the required slew angle, then the satellite would have rotated through an angle of $\frac{1}{2}\theta_f$ after $\frac{1}{2}t_f$. The above-mentioned concept is illustrated in Figure 3.2, where $\theta(t)$ is the rotation angle, $\dot{\theta}(t)$ is the angular rate and $\ddot{\theta}(t)$ is the angular acceleration as a function of time t.
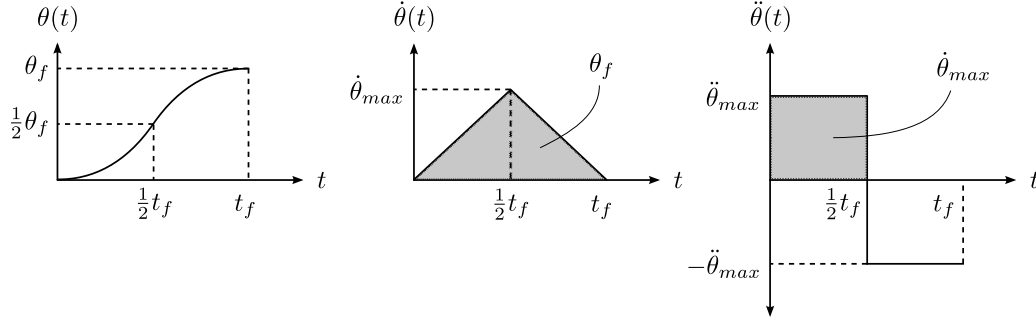
Figure 3.2 – The angle, angular rate, and angular acceleration for a Bang-Bang control strategy

From the shaded areas in Figure 3.2 it can be deducted that

$$\theta_f \;=\; \frac{1}{2}t_f\dot{\theta}_{max} \tag{3.4.1}$$

$$\text{and}\quad \dot{\theta}_{max} \;=\; \frac{1}{2}t_f\ddot{\theta}_{max}\;. \tag{3.4.2}$$

$\ddot{\theta}_{max}$ can be expressed in terms of θ_f by combining Equations 3.4.1 and 3.4.2, which delivers

$$\ddot{\theta}_{max} \;=\; \frac{4\theta_f}{t_f^2}\;. \tag{3.4.3}$$

$N_{ws,max}$ is furthermore a function of $\ddot{\theta}_{max}$ through

$$N_{ws,max} \;=\; \ddot{\theta}_{max}I_{max}\;, \tag{3.4.4}$$

where I_{max} is the satellite's largest MoI. $h_{ws,max}$ can similarly be calculated via

$$h_{ws,max} \;=\; \dot{\theta}_{max}I_{max}\;. \tag{3.4.5}$$

Implementing Equations 3.4.2 to 3.4.5 for the 20 kg satellite (which has $I_{max} = 0.45$ kgm^2) delivers

$$\dot{\theta}_{max} \;=\; 2\,°/\text{s}\;,$$

$$\ddot{\theta}_{max} \;=\; 0.133\,°/\text{s}^2\;,$$

$$h_{ws,max} \;=\; 15.7\text{ mNms}\;,$$

$$\text{and}\quad N_{ws,max} \;=\; 1.05\text{ mNm}\;.$$

3.5   Maximum Disturbance Torques

The reaction wheels of the satellite must not only be able to bring about the desired agility, but also be able to overcome the disturbance torques experienced by the satellite. Other actuators are therefore also required to dump the angular momentum built up in the reaction wheels. The major sources of disturbance torques are gravity gradient, atmospheric drag, solar radiation pressure (SRP), and the satellite's residual magnetic moment [9].

The maximum magnitude N_{gg} of the gravity gradient disturbance torque for the 20 kg satellite can be approximated as

$$N_{gg} = \frac{3\mu_E}{2(R_E + h)^3} |I_{zz} - I_{xx}| \sin(2\theta) , \qquad (3.5.1)$$

where h is the satellite's altitude and θ is the pitch angle.

The maximum magnitude N_a of the atmospheric drag disturbance torque is given by

$$N_a = \frac{1}{2}\rho C_d A_v V^2 c_{ap-m} , \qquad (3.5.2)$$

where ρ is the atmospheric density, C_d is the satellite's drag coefficient, A_v is the satellite's projected area in the velocity direction, V is the satellite's velocity, and c_{ap-m} is the magnitude of the centre of atmospheric pressure to centre of mass offset vector.

The maximum magnitude N_s of the SRP disturbance torque can be expressed as

$$N_s = \frac{\Phi}{c} A_s (1 + q) c_{sp-m} \cos(\varphi) , \qquad (3.5.3)$$

where Φ is the solar constant, c is the speed of light, A_s is the projected area in the direction of the sun, q is the satellite's reflectance factor, c_{sp-m} is the magnitude of the centre of solar radiation pressure to centre of mass offset vector, and φ is the sun incidence angle.

The maximum magnitude N_{rm} of the residual magnetic moment disturbance torque is given by

$$N_{rm} = DB_{max} , \qquad (3.5.4)$$

where D is the satellite's residual magnetic moment and B_{max} is the maximum magnitude of the geomagnetic field vector.

Table 3.2 summarises the calculation of N_{gg}, N_{atm}, N_s, N_{rm} for the 20 kg satellite. Where possible, worst case values [9] were assumed.

Table 3.2 indicates that each reaction wheel would need to be able to generate 5.99 μNm to overcome the disturbance torques. This value is the absolute worst case, since it is extremely unlikely that all

Source	Assumptions	Torque (μNm)
N_{gg}	$h = 500$ km $\mid \theta = 45°$	0.28
N_a	$\rho = 2 \times 10^{-12}$ kg/m^3 $\mid C_d = 2.5 \mid A_v = 0.12$ m^2 $\mid c_{ap-m} = 0.05$ m	0.87
N_s	$\Phi = 1368$ W/m^2 $\mid A_s = 0.12$ m^2 $\mid q = 0.6 \mid c_{sp-m} = 0.05$ m $\mid \varphi = 0°$	0.04
N_{rm}	$D = 0.1$ Am^2 $\mid B_{max} = 48\mu$T	4.8
Total	$-$	5.99

Table 3.2 – Maximum disturbance torques

the disturbance torques would be in the same direction (i.e. about the same axis). The orbit period T_o at and altitude of 500 km is known to be 94.62 minutes [9], or 5677.2 s. The worst case maximum angular momentum build-up h_d over one orbit as a result of the total disturbance torque N_d can therefore be calculated as

$$
\begin{aligned}
h_d &= N_d T_o \\
&= 34 \text{ mNms .}
\end{aligned}
$$
(3.5.5)

3.6   ADCS Hardware

At this point in the initial ADCS analysis the hardware should be chosen.

Except for the high accuracy sensor that is required during imaging, a sun sensor (or sun sensors) is required if sun tracking is to be performed. Determining the sun position using the solar panels is not always feasible, since solar panels will not necessarily be on all the facets of the satellite. Both a coarse sun sensor (CSS) and a fine sun sensor (FSS) were chosen for the 20 kg satellite to enable coarse sun tracking during de-tumbling and fine sun tracking during normal operation. A 3-axis magnetometer was also added to enable magnetic control, which is to be used during de-tumbling and for managing the reaction wheel angular momentum.

Although star trackers are traditionally only used on larger satellites [14], many star trackers have recently been developed for use on smaller satellites. Examples include the star trackers from ComTech AeroAstro, Sinclair Interplanetary, Electronic Systems Laboratory (ESL), Blue Canyon Technologies, and Berlin Space Technologies. These advances bring forth the ability to achieve precision pointing on smaller satellites and a star tracker was thus chosen to provide accurate measurements during imaging.

As mentioned before, 3-axis magnetic control will be used to manage the wheel angular momentum. The magnetic torquers must thus be able to generate enough torque to overcome the disturbance torques that will cause an angular momentum build-up in the wheels. The absolute worst case angular momentum build-up for a single wheel for one orbit was calculated using Equation 3.5.5 as $h_d = 34$ mNms. Since high accuracy and stability can be required for imaging at any time during the sunlit part of the orbit, it was chosen that angular momentum management will only be performed during

eclipse. For a 500 km orbit, the maximum eclipse period T_e is 2145 s [9], which means that the required magnetic torque $N_{m,req}$ can be calculated as

$$
\begin{aligned}
N_{m,req} &= \frac{h_d}{T_e} \\
&= 15.9 \ \mu\text{Nm} \ .
\end{aligned}
\tag{3.6.1}
$$

If the minimum geomagnetic field vector magnitude B_{min} is assumed to be 25 μT, then the minimum required magnetic torquer dipole moment D_{min} is calculated as

$$
\begin{aligned}
D_{min} &= \frac{N_{m,req}}{B_{min}} \\
&= 0.64 \ \text{Am}^2 \ .
\end{aligned}
\tag{3.6.2}
$$

A commercial-off-the-shelf product such as the MT1-1 from ZARM Technik (with a maximum dipole moment of 1 Am^2) can be used as the X-axis, Y-axis and Z-axis magnetic torquers.

The minimum required angular momentum of the reaction wheels will be affected by both the agility requirement and the disturbance-torque-induced angular momentum build-up. The angular momentum build-up h_{ds} during the sunlit part of the orbit (which has a duration of $T_o - T_e$) can be calculated as

$$
\begin{aligned}
h_{ds} &= h_d \left(\frac{T_o - T_e}{T_o} \right) \\
&= 21.2 \ \text{mNms} \ .
\end{aligned}
\tag{3.6.3}
$$

The required angular momentum storage capability $h_{w,req}$ of the reaction wheels can thus be calculated as

$$
\begin{aligned}
h_{w,req} &= h_{ws,max} + h_{ds} \\
&= 36.9 \ \text{mNms} \ .
\end{aligned}
\tag{3.6.4}
$$

The torque requirement $N_{w,req}$ of the reaction wheels can similarly be calculated as

$$
\begin{aligned}
N_{w,req} &= N_{ws,max} + N_d \\
&= 1.06 \ \text{mNm} \ .
\end{aligned}
\tag{3.6.5}
$$

A commercial-off-the-shelf product such as the RW-0.06 from Sinclair Interplanetary (with $h_{w,max} = 60$ mNms and $N_{w,max} = 5$ mNm) can be used as the X-axis, Y-axis, and Z-axis reaction wheels.

3.7 Jitter

Once the reaction wheels have been chosen, a jitter analysis is required to ensure that the stability requirement from Section 3.2 is met. Any reaction wheel is subject to a number of disturbances which will all influence the stability of the satellite during fine-pointing. The disturbance sources that were considered for analysis included wheel imbalance, motor torque ripple, and wheel speed resolution. Furthermore, the satellite body axis with the smallest moment of inertia will experience the most jitter. As a result, only the worst case jitter about the Z-axis will be analysed in this section.

3.7.1 Wheel Imbalance

Wheel imbalance is specified in terms of static imbalance U_s and dynamic imbalance U_d, as discussed in Chapter 2. For the RW-0.06, these values are given as $U_s = 0.0208$ gcm and $U_d = 0.0208$ gcm^2 and the moment of inertia of the wheel is given as $I_w = 88.1 \times 10^{-6}$ kgm^2. With the absolute worst case required angular momentum capability of the reaction wheels being $h_{w,req} = 36.9$ mNms, the worst case maximum Z-wheel speed ω_w will reach

$$
\begin{aligned}
\omega_w &= \frac{h_{w,req}}{I_w} \\
&= 418.8 \text{ rad/s}
\end{aligned} \tag{3.7.1}
$$

before the magnetic torquers will start dumping the angular momentum. As mentioned in Chapter 2, the static imbalance torque will be a function of U_s, ω_w, and D_w, the distance from the Z-wheel's axis of rotation to the satellite body Z-axis, or Z_B. D_w was assumed to be a maximum of 0.1 m for this project. The magnitude of the imbalance force F_s (determined from Equation 2.3.7) in the XY-plane can be used to calculated the magnitude of the satellite body angular acceleration $\dot{\omega}_{zs}$ (as a result of the static imbalance) as

$$
\begin{aligned}
|\dot{\omega}_{zs}| &= \frac{D_w|F_s|}{I_{zz}} \\
&= 0.012 \text{ rad/s}^2 \ ,
\end{aligned} \tag{3.7.2}
$$

where $D_w|F_s|$ represents the magnitude of the static imbalance torque about Z_B. Since $\dot{\omega}_{zs}$ is the time derivative of the static imbalance angular rate ω_{zs}, the magnitude of ω_{zs} will be smaller than that of $\dot{\omega}_{zs}$ by a factor of ω_w, which is the frequency of $\dot{\omega}_{zs}$. The magnitude of ω_{zs} can thus be calculated as

$$
\begin{aligned}
|\omega_{zs}| &= \frac{|\dot{\omega}_{zs}|}{\omega_w} \\
&= 5.99 \text{ arcsec/s} \ .
\end{aligned} \tag{3.7.3}
$$

The dynamic imbalance U_d of the X-wheel and the Y-wheel will cause a disturbance torque about the Z-axis. If the worst case angular rate of $\omega_w = 418.8$ rad/s for one of the wheels is once again assumed,

the magnitude of the angular acceleration $\dot{\omega}_{zd}$ about the Z-axis (as a result of the dynamic imbalance torque) can be expressed as

$$
\begin{aligned}
|\dot{\omega}_{zd}| &= \frac{U_d\,\omega_w^2}{I_{zz}} & (3.7.4) \\
&= 0.0012 \text{ rad/s}^2 \ .
\end{aligned}
$$

The magnitude of the dynamic imbalance angular rate jitter ω_{zd} can therefore be calculated as

$$
\begin{aligned}
|\omega_{zd}| &= \frac{|\dot{\omega}_{zd}|}{\omega_w} & (3.7.5) \\
&= 0.6 \text{ arcsec/s} \ .
\end{aligned}
$$

3.7.2   Motor Torque Ripple

The commutation of the motor driving the reaction wheel causes a torque ripple, which translates to an angular rate jitter. The frequency of the jitter is the commutation rate multiplied by the wheel speed. If it is assumed that the worst case torque ripple is 25% of the maximum wheel torque and that the motor has $p = 6$ poles, then the magnitude of the Z-axis torque-ripple-induced angular rate jitter $\omega_{z,ripple}$ can be calculated as

$$
\begin{aligned}
|\omega_{z,ripple}| &= \frac{0.25 N_{w,max}}{I_{zz}\omega_w p} & (3.7.6) \\
&= 0.34 \text{ arcsec/s} \ .
\end{aligned}
$$

3.7.3   Wheel Speed Resolution

The speed of the reaction wheel is controlled by electronics that is subject to quantisation. The process of quantisation translates to a certain wheel speed resolution, which will cause an angular rate jitter $\omega_{z,res}$ about the respective satellite body axis (in this case the Z-axis). If it is assumed that the wheel speed resolution is $\Delta\omega_w = 0.5$ rpm, then the magnitude of ω_{res} becomes

$$
\begin{aligned}
|\omega_{res}| &= \Delta\omega_w\left(\frac{I_w}{I_{zz}}\right) & (3.7.7) \\
&= 3.17 \text{ arcsec/s} \ .
\end{aligned}
$$

3.7.4   Total Jitter

The total worst case Z-axis jitter $\omega_{z,jitter}$ can be calculated from the individual jitter components as

$$
\begin{aligned}
\omega_{z,jitter} &= |\omega_{zs}| + |\omega_{zd}| + |\omega_{z,ripple}| + |\omega_{res}| & (3.7.8) \\
&= 10.1 \text{ arcsec/s} \ .
\end{aligned}
$$

The worst case angular rate jitter is therefore well within the stability requirement of 72.8 arcsec/s.

3.8   Summary

An initial analysis of the 20 kg satellite's ADCS was shown in this chapter. The specifications and requirements of the satellite were firstly given, after which the requirements were translated to sensor and actuator requirements.

A star tracker was chosen to satisfy the position knowledge requirement and a coarse and fine sun sensor were added to allow coarse and fine sun tracking to be performed. A magnetometer was also included to enable magnetic control on the satellite.

The pointing accuracy requirement will be satisfied using three reaction wheels, of which the angular momentum will be managed using magnetic torquers. The angular momentum and torque specifications of the chosen reaction wheels are of such a nature that the wheels will be able to perform the required agility slew while also compensating for various disturbance torques. The chosen magnetic torquers will furthermore be able to dump the wheels' angular momentum build-up and the momentum dumping procedure was chosen to be executed during eclipse only.

The worst case jitter that would be caused by any one of reaction wheels was lastly calculated and it was found that the angular rate jitter will be well within the required stability.

Chapter 4

An Application Specific Simulation Setup

4.1 Overview

Any accurate ADCS simulation should be based on a given set of application specific parameters and models, ranging from the satellite's orbit to the reaction wheel speed measurement noise. This chapter will describe the construction of application specific simulation models based on the orbit and hardware chosen for the 20 kg EO satellite.

The orbit of the satellite, as well as the position of the imager payload, communication payload, and main solar panel with respect to the satellite body, will firstly be defined. Knowledge of the aforementioned positions is required by the ADCS when performing imaging or target tracking manoeuvres, as well as during sun-seeking periods. Thereafter the development of simulation models of the ADCS sensors and actuators selected in Chapter 3 will be discussed. Without these models, the simulation environment would produce unrealistic results. A practical example of the process of deriving a reaction wheel model will given be in conclusion of this chapter.

4.2 Orbit Parameters

The orbit of the 20 kg EO satellite is a 500 km sun-synchronous orbit. A slightly modified TLE of a known SSO satellite served as the platform for SGP4 orbit propagation. The following parameters were extracted from the modified TLE:

- Epoch: 2014/08/01, 03:01:16

- Inclination (i): 97.4°

- RAAN (Ω): 275.0°

- Argument of perigee (ω): 57.4°

- Eccentricity (e): 0.000092

- Mean anomaly (M): 29.3°

- Mean motion (n): 15.2355 revolutions/day

- Drag term (B^*): 0.000194

The ADCS simulation results that will be shown in Chapters 5 to 7 were all based on the above-mentioned parameters.

4.3   Satellite Body

Several factors regarding the physical satellite body structure affect the ADCS in terms of sensor placement and control commands. For example, the SBC vector pertaining to the imager boresight on an EO satellite needs to be defined before the ADCS can point the payload towards a desired target. For the project at hand, the imager's SBC boresight unit vector $\mathbf{u}_{bs}^{SBC}$ was chosen in the direction of the positive body Z-axis, thus

$$\mathbf{u}_{bs}^{SBC} \;=\; \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T . \tag{4.3.1}$$

The unit vector $\mathbf{u}_{sp}^{SBC}$, which indicates the main solar panel's position, needed to be defined to allow the ADCS to perform coarse and fine sun tracking. Given the fact that the orbit described in Section 4.2 is approximately a 10PM/10AM SSO, the solar panel was chosen to be on the $-$Y-facet of the satellite. The last-mentioned implies that if the satellite is in a nominal attitude (i.e. when the SBC frame is aligned with the ORC frame), the solar panel will be on the sun-facing side of the orbit plane. The sun incidence angle on the solar panel will be roughly 60° for the nominal orientation, which means that the angle β_{sun} between the sun and the orbit plane is approximately 30°. $\mathbf{u}_{sp}^{SBC}$ is therefore defined as

$$\mathbf{u}_{sp}^{SBC} \;=\; \begin{bmatrix} 0 & -1 & 0 \end{bmatrix}^T . \tag{4.3.2}$$

It is lastly also assumed that the communication payload is on the +Z-facet of the satellite.

4.4   Sensor Models

All ADCS sensors have at least one thing in common: noise. Various sources affect the level and spectrum of the noise, namely high frequency measurement inaccuracy, bias drift, structural alignment, and digital quantisation. Only high frequency noise was chosen to be modelled the simulation environment. It was thus assumed that low frequency noise sources such as bias drift are negligible and that the sensor calibration with respect to structural alignment has already been accomplished.

It was further assumed that the high frequency noise added to the sensor measurements is zero-mean Gaussian noise. In some cases the standard deviation of the noise was based on that of a commercial-off-the-shelf (COTS) sensor. The sensor models are all implemented similarly with respect to measurement noise. An ORC reference vector (three vectors in the case of the star tracker) is generated and transformed to SBC using the DCM $\mathbf{A}$, after which the measurement noise is added to each component of the vector. If $\mathbf{v}_o$ is the reference ORC vector and $\mathbf{m}_v$ is the measurement noise, then the measured SBC vector $\mathbf{v}_b$ is calculated as

$$\mathbf{v}_b \;\;=\;\; \mathbf{A}\mathbf{v}_o + \mathbf{m}_v \; . \tag{4.4.1}$$

The placement of the sensors on the satellite body is to a great extent application specific, especially with respect to the primary payload and the desired orbit. The sections following hereafter will therefore also discuss the placement of the ADCS sensors on the body of the 20 kg satellite.

4.4.1   Magnetometer

The measured and IGRF-modelled geomagnetic field vectors $\mathbf{B}_b$ and $\mathbf{B}_o$ can be used to determine the attitude and angular rates of the satellite [15] and $\mathbf{B}_b$ can furthermore be used in the magnetic attitude control process. The standard deviation (or σ) of the measurement noise was chosen as 0.5 μT per axis, which means that the maximum error in $\mathbf{B}_b$ would be approximately 1.5 μT per axis. Figure 4.1 shows an example of the measured geomagnetic field vector in SBC.
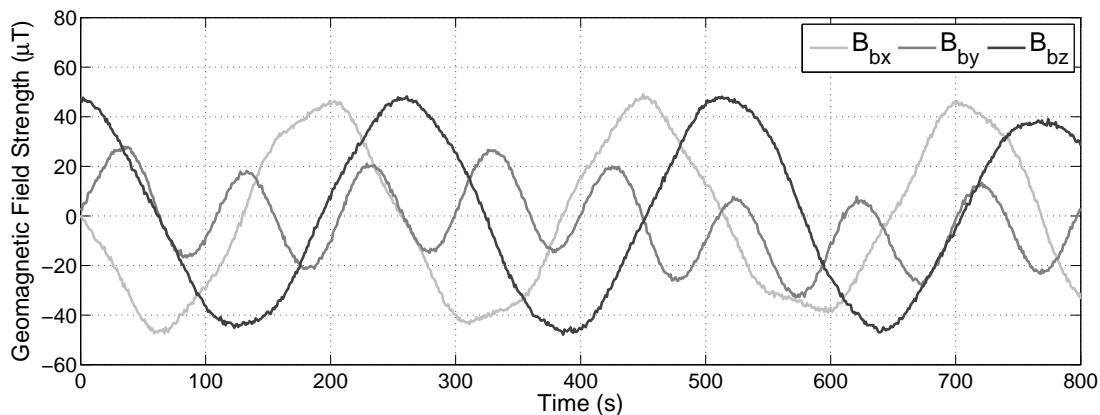
Figure 4.1 – Modelled magnetometer measurements

The magnetometer is also the only ADCS sensor for this project that is able to provide measurements at any given attitude and orbit position, implying that its placement on the satellite body does not need to be specified for simulation purposes.

4.4.2   Sun Sensors

The coarse sun sensor model in the simulation environment is based on a simple photodiode array. This type of CSS consists of at least 6 photodiodes mounted on all the facets of the satellite, each producing a level of light intensity from which a coarse unit sun vector can be constructed. Although a photodiode-array-CSS is not very accurate, its low cost makes it attractive for small satellite application.

With every facet of the satellite containing at least one photodiode, a CSS measurement will always be available when the satellite is in the sunlit part of the orbit, regardless of the attitude. The measurement noise was chosen to allow a maximum error in $\mathbf{S}_b$ of $9°$. Figure 4.2 shows an example of unit sun vector in SBC, as measured by the CSS during the sunlit part of the orbit.
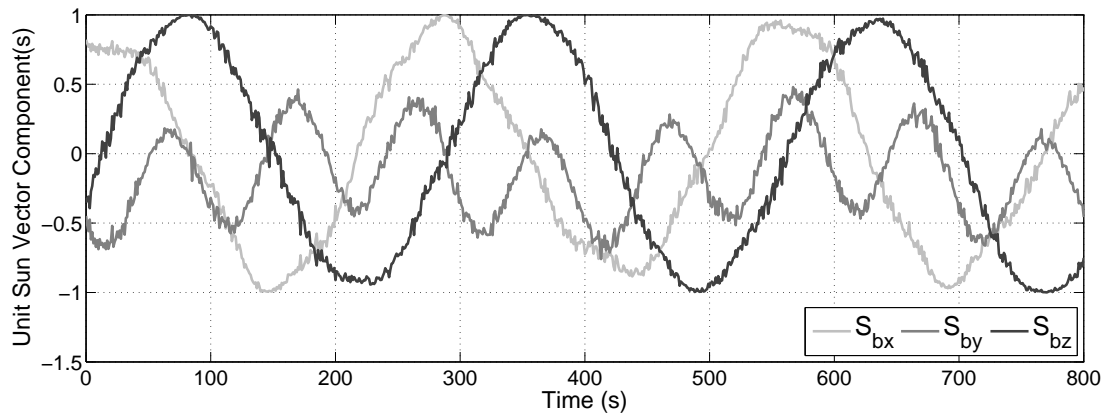
Figure 4.2 – Modelled coarse sun sensor measurements

The implementation of the fine sun sensor model is more complex. The FOV of the sensor must firstly be brought into consideration, which means that the position of the FSS on the satellite body must be defined. The FSS was chosen to be mounted on the $-Y$-facet of the satellite, which implies that the sun will be in the sensor's FOV when the satellite is in the nominal orientation (and the sensor has a FOV radius larger than $90° - \beta_{sun}$). The FOV of the FSS is based on CubeSense, a COTS FSS manufactured by the Electronic Systems Laboratory (ESL) at Stellenbosch University in South Africa, which has FOV of $180°$ (thus a half sphere). The measurement noise standard deviation σ for the chosen FSS is a function of incidence angle. If β_{FSS} is the sun incidence angle on the FSS, then the chosen σ can be expressed as

$$\sigma = 0.1° \quad \text{if } \beta_{FSS} \leq 40°$$
$$\sigma = 0.3° \quad \text{if } \beta_{FSS} \leq 60°$$
$$\sigma = 0.5° \quad \text{if } \beta_{FSS} \leq 90° \ .$$

It should also be noted that the FSS measurement is invalid if the incidence angle is more than $90°$. Figure 4.2 shows an example of FSS measurements during the sunlit part of the orbit.
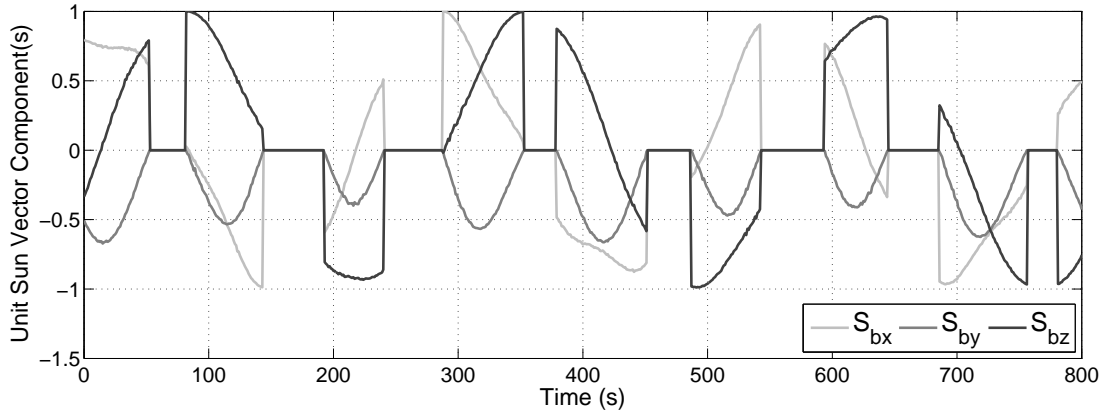
Figure 4.3 – Modelled fine sun sensor measurements

4.4.3   Star Tracker

The star tracker in the ADCS simulation environment is based on a model similar to that of the magnetometer and the sun sensors. In the case of the star tracker however, three ORC vectors are generated. The last-mentioned is driven by the fact that a star tracker requires at least three measured stars to produce an attitude solution [14]. The three ORC vectors are once again transformed to SBC using the DCM **A**, after which noise is added to produce simulated measurements.

The nature of the star tracker measurement noise differs from that of the other sensors. Although a star tracker's noise is typically expressed in degrees (or arcseconds), the star tracker's accuracy is directly related to how accurate it can determine the centroids of stars in terms of pixels [14]. A pixel on the star tracker's image sensor has a certain length and width (typically in the μm range), thus the centroid of a star can also be expressed in mm as a point on a two-dimensional image plane. The measurement noise is therefor a centroid location error (in mm) on the image plane. With the focal length of the star tracker adding the third dimension, the measured centroid can be converted to a three-dimensional measurement vector in mm. An illustration of the above-mentioned concept can be seen in Figure 4.4, where the XY-plane is parallel to the image plane, the Z-axis represents the star tracker's boresight, f_{ST} is the focal length, and x_s and x_y represent the calculated centroid position.

The measurement noise of a star tracker is often specified in terms of across-boresight noise and around-boresight noise, the latter being the larger of the two. The star tracker model in the simulation environment is based on CubeStar, a COTS star tracker with a focal length of $f_{ST} = 6.28$, mm developed by the ESL. The 1-σ across-boresight accuracy is assumed to be a worst case of $\sigma_{deg} = 0.02°$, which means that σ_{mm}, the 1-σ centroid error (in mm) on the image plane, can be calculated as

$$
\begin{aligned}
\sigma_{mm} &= f_{ST} \tan\left(\sigma_{deg}\right) & (4.4.2)\\
&= 6.28 \tan\left(0.02°\right)\\
&= 0.0022 \text{ mm .}
\end{aligned}
$$
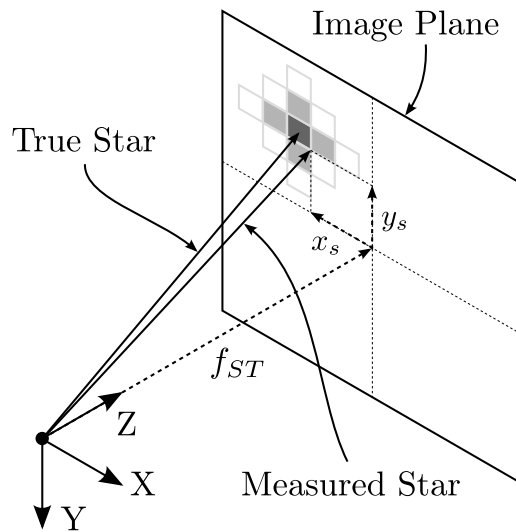
Figure 4.4 – An illustration of the star tracker model

With CubeStar's pixel size of 5.6μm $\times$ 5.6μm, the centroid location error is approximately 0.4 pixels.

The three ORC reference vectors are generated through three reference centroids (in mm on the image plane) that are converted to three-dimensional vectors using f_{ST}. It is furthermore assumed that the star tracker will always be able to match at least three stars. The last-mentioned implies that if neither the Earth nor the sun is in the star tracker's FOV, three measured vectors will be produced. The FOV of the star tracker was furthermore chosen to be circular with a radius of $20°$.

The star tracker was chosen to be mounted between the $+$Y-axis and the $-$Z-axis at an angle of $45°$ (with respect to the SBC frame). In other words, if the satellite is in its nominal attitude, the star tracker will be pointed $45°$ upwards from the orbit $+$Y-axis in the YZ plane. The chosen mounting position ensures that neither the sun nor the Earth will come into the star tracker's FOV during imaging, even in the presence of a $30°$ roll offset angle (and $0°$ pitch and yaw angles).

4.5   Actuator Models

Although the internal dynamics of ADCS actuators are ignored by the ADCS control laws, the response of an actuator to a given command (or reference) is important when aiming to simulate a satellite's attitude accurately. In practice, actuators are subject to among others digital quantisation, physical constraints, and inner closed loop dynamics. This section will show how the simulation environment models for the magnetic torquers and the reaction wheels were created. The practical example that served as the foundation of the reaction wheel model will also be discussed.

4.5.1 Magnetic Torquers

Magnetic torquers are often employed for coarse attitude control and as mentioned in Chapter 1, most satellites will employ some form of magnetic control during its lifetime. A magnetic torquer consists of a coil with a ferromagnetic core (although air-core torquers are also common) that is driven by some form of electronics. Although in practice the "discrete nature of the on-board control processors" lead to the adoption of a pulse width modulation (PWM) control technique [16], the ADCS simulation environment employs a continuous signal to simulate the generated magnetic dipole moment **M**.

The magnetic controller in the simulation environment produces a commanded dipole moment of which the components are then scaled by the torquers' maximum-on-time-to-sample-period ratio. For example, if the controller commands the torquer to generate a dipole moment of 2 Am^2 for 900 ms, the actual output in the simulation will be 1.8 Am^2 for 1 s. The magnetic torquers modelled in the ADCS simulation environment of this project are based on an 80% maximum on-time scheme. The remaining 20% off-time symbolises the magnetometer's measurement window. An on-time resolution of 10 ms is also implemented to represent digital quantisation. Therefore, with a chosen magnetic control sample period of 1 s, the amount of torquer on-time resolution steps is $\frac{80\% \times 1\text{s}}{10\text{ms}} = 80$.

4.5.2 Reaction Wheels

Reaction wheels are an integral part of an EO satellite's ADCS. In most cases, RWs are employed when stability and precision pointing are required, for example during an imaging manoeuvre. The dynamics associated with a reaction wheel can affect the performance of the ADCS to a great extent. The last-mentioned created the need for the implementation of a relatively accurate RW model in the ADCS simulation environment. An engineering model RW will be used as an example in the process of deriving such a model.

The size of the engineering model RW is similar to that of the RW chosen in Chapter 3. The motor is driven by a driver integrated circuit (IC) that receives 10-bit PWM commands from a microcontroller. The angular rate (or speed) of the RW is measured by a 12-bit magnetic encoder, which sends its measurements back to the microcontroller every 5 ms. A 95% recursive average filter was applied to the measurements to suppress high frequency noise (more on this later).

An ideal high-level model of a RW would entail immediate speed response. However, this is not the case in practice. The modus operandi when developing a satellite reaction wheel is to determine the open loop response of the reaction wheel and to design a discrete controller to meet the desired closed loop specifications. The open loop response of a motor is generally first order with a time constant τ. Figure 4.5 shows a few examples of the RW's open loop response (for various duty cycles D), verifying the first order assumption.

Figure 4.5 indicates that the time constant (i.e. the time at which 63.2% of the steady state value is reached) of the RW is roughly 8 s. The inner speed control loop was chosen to have a sample period
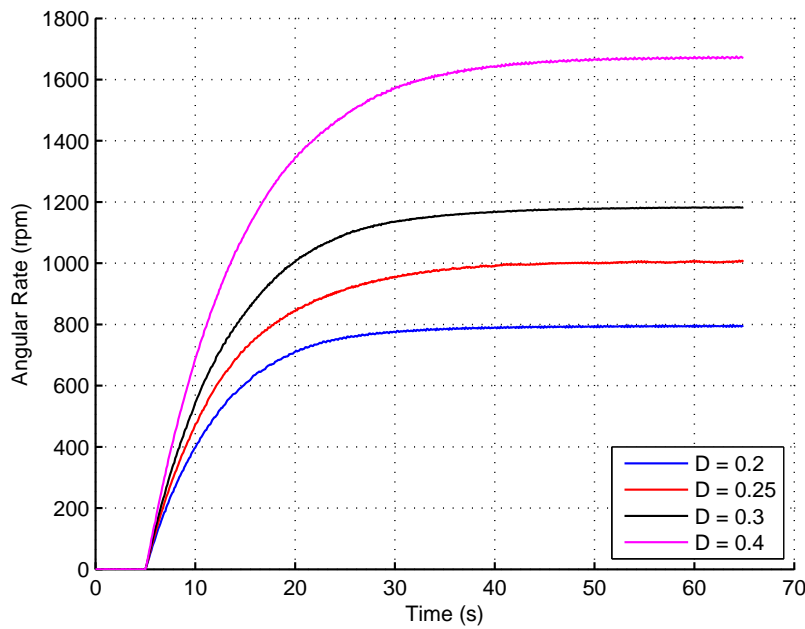
Figure 4.5 – Open loop step responses of the RW

of $T_w = 100$ ms, which is orders of magnitude smaller than τ. The last-mentioned led to the plant $G(s)$ of the RW being approximated as an integrator (or a first order system with an infinite time constant). Otherwise stated, the controller sample period is so small that the controller will only "see" the linear part of the RW's open loop response. If K_s is the integrator gain, then

$$G(s) \quad = \quad \frac{K_s}{s} \; . \tag{4.5.1}$$

Since the controller was to be implemented on a digital platform, $G(s)$ needed to be converted to a discrete domain equivalent $G(z)$. Using a zero-order hold (ZOH) function, $G(z)$ was calculated as

$$G(z) \quad = \quad \frac{K_s T_w}{z - 1} \; . \tag{4.5.2}$$

The integrator gain K_s is dependent on the input and output units of the plant. The input of the RW was chosen as the digital PWM register units and the output was chosen as speed in rpm, which means that K_s will have the units angular-acceleration-per-digital-unit. To avoid non-linearities near zero speed, the gain K_s was calculated by giving the RW various PWM commands from an offset speed. A 25% duty cycle was applied to the RW and the speed was allowed to reach steady state, after which six different duty cycle commands (which are equivalent to PWM commands) were given. Figure 4.6 shows the resulting responses and Figure 4.7 shows the linear regions of these responses, which will be used to determine K_s. Figure 4.7 furthermore confirms the validity of the integrator plant approximation.

The first step in calculating K_s was to calculate $\frac{\Delta \omega}{\Delta t}$ over a chosen period of 1 s. K_s was then calculated for each case using the appropriate PWM register value (or PWM_{reg}). After discarding the highest
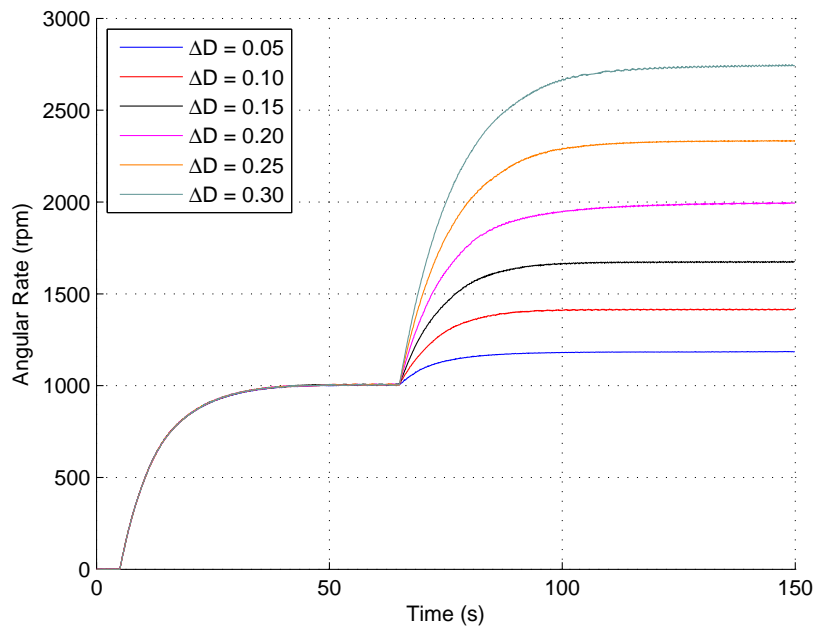
Figure 4.6 – Open loop step responses of the RW from a speed bias
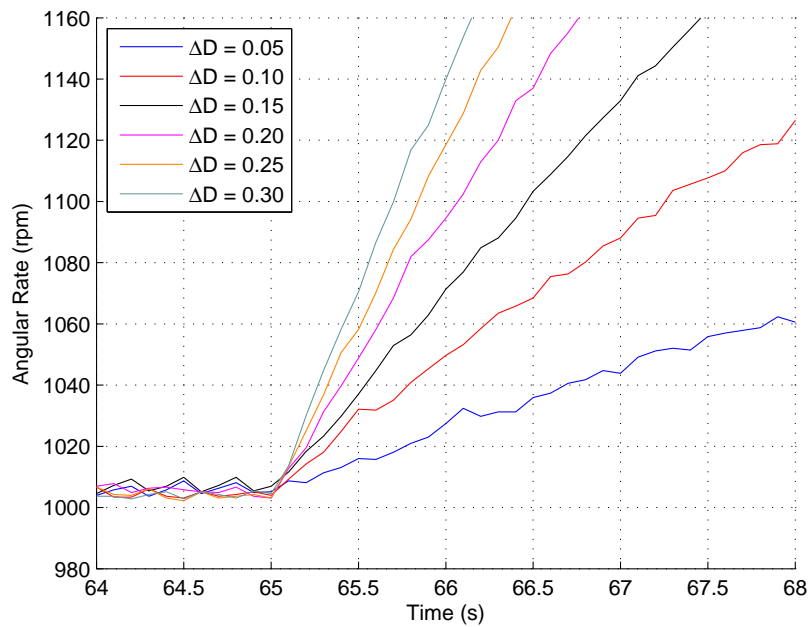
Figure 4.7 – An illustration of the RW integrator plant approximation

and lowest of the six calculated K_s values, the average of the remaining values was calculated as $K_s = 0.443$. Table 4.1 illustrates the above-mentioned process of calculating K_s.

The dynamics of the added measurement filter also needed to be added to the plant before the controller

$\mathbf{\Delta D}$	$\frac{\mathbf{\Delta \omega}}{\mathbf{\Delta t}}$	$\mathbf{PWM_{reg}}$	$\mathbf{K_s}$
$-$	[rpm/s]	D/A units	$\frac{\text{rpm/s}}{\text{D/A units}}$
0.05	22.27	51	0.437
0.10	45.70	102	0.448
0.15	64.45	153	0.421
0.20	91.41	205	0.446
0.25	115.43	256	0.451
0.30	135.36	307	0.441

Table 4.1 – The calculation of the open loop RW gain K_s

could be designed. The 95% recursive filter $M(z)$ can be expressed as

$$M(z) \;=\; \frac{K_m z}{z - m} \;, \tag{4.5.3}$$

where $K_m = 1 - m$ and $m = 0.95$. The sample period of two discrete plants must however be equal before they can be combined. $M(z)$ thus needed to be converted to a sample period of 100 ms, which was achieved through

$$
\begin{aligned}
M(z) \;&=\; \frac{(1 - 0.95^{\frac{100}{5}})z}{z - 0.95^{\frac{100}{5}}} \\
&=\; \frac{0.6145z}{z - 0.3585} \\
&=\; \frac{K_m z}{z - m} \;. \tag{4.5.4}
\end{aligned}
$$

The complete open loop transfer function of the wheel and the sensor $G_w(z)$ is thus given by

$$G_w(z) \;=\; \frac{K_s K_m T_w z}{(z - 1)(z - m)} \;. \tag{4.5.5}$$

The next step in constructing the RW model was designing the discrete controller $D(z)$. The dominant second order closed loop poles of the system were chosen to be optimally damped with a settling time 20 times larger than the sample period. A higher settling time would yield a response that is undesirably slow, while a lower settling might generate control noise. The desired closed loop poles ($z_{cl,1}$ and $z_{cl,2}$) could therefore be calculated as

$$
\begin{aligned}
z_{cl,1} \;&=\; 0.781 + j0.176 \\
\text{and} \quad z_{cl,2} \;&=\; 0.781 - j0.176 \;.
\end{aligned}
$$

$D(z)$ required an integrator to force the steady state error for step commands to zero. The integrator of the RW plant alone will not ensure a zero steady state error, since the approximated plant is

subject to disturbance signals. A proportional-integral (PI) controller and a proportional-integral-derivative (PID) controller were investigated for $D(z)$. Figure 4.8 shows the simulated response of both controllers. The larger overshoot for the PI controller is a result of a third closed loop pole which is closer to the dominant poles than in the case of the PID controller. The PID controller was thus selected as the preferred controller.
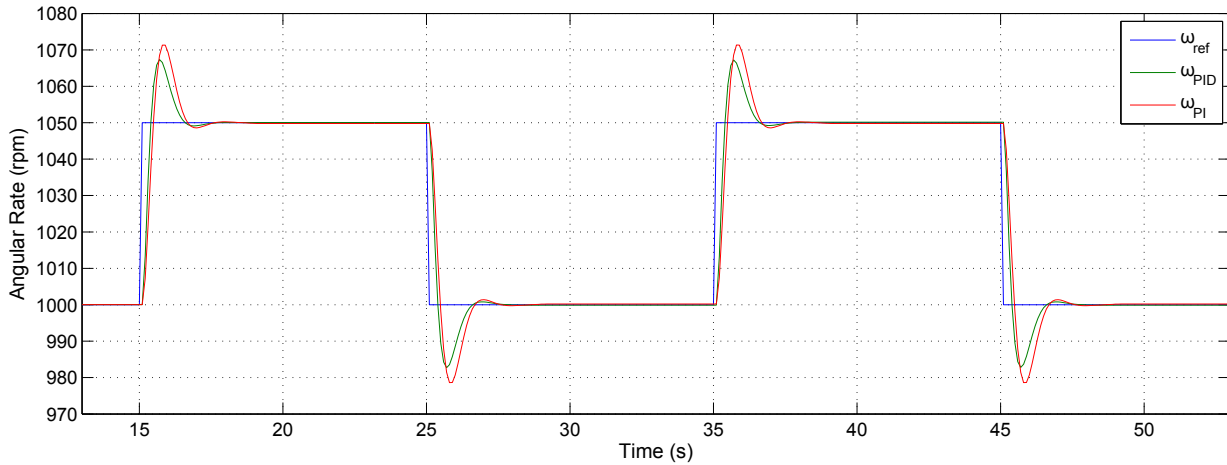
Figure 4.8 – A comparison of RW speed controllers using step responses

$D(z)$ was implemented on the microcontroller as a difference equation, which was calculated as

$$u(k) \quad = \quad u(k-1) + 15.404\ e(k) - 18.144\ e(k-1) + 4.525\ e(k-2)\ , \tag{4.5.6}$$

where $u(k)$ is the control signal (i.e. the PWM register value, which is the integer magnitude of $u(k)$, and the direction, which is the sign of $u(k)$) and $e(k)$ is the speed error.

Three tests were set up to investigate the performance of the PID controller, namely a step response test, a steady state test, and a ramp response test. Figure 4.9 shows measured and simulated step responses of the RW for the first test, from which it is evident that the closed loop system responded as expected.

Figures 4.10 and 4.11 show the steady state performance of the PID controller during the second test. The reference speed was set to 1000 rpm, 1050 rpm, 1100 rpm, 2000 rpm, 2050 rpm, and 2100 rpm respectively for 60 seconds at each rate.

An analysis of Figures 4.10 and 4.11 provided several interesting findings, which are summarised in Table 4.2. Firstly, offsets were apparent between the reference angular rate (ω_{ref}) and the mean of the measured angular rate (ω_{meas}) for each case. The standard deviation of the angular rate (σ_{meas}) can be seen to have varied for each case, although the variation was not necessarily a function of ω_{ref}. The mean of the rate error ($\overline{\omega}_{error}$), which is the difference between the reference rate and the measured rate, is also shown in Table 4.2. Since a mean offset between ω_{ref} and ω_{meas} was evident,
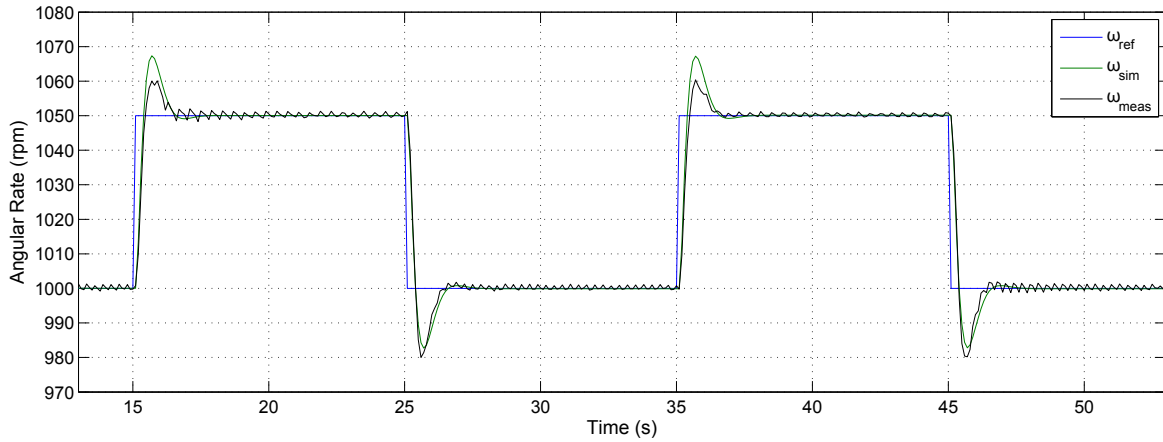
Figure 4.9 – The step response RW test
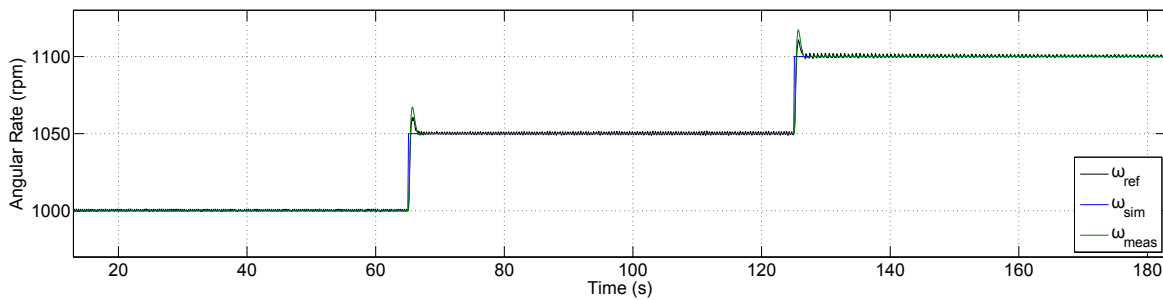
Figure 4.10 – The steady state RW test ($\omega_{bias} = 1000$ rpm)
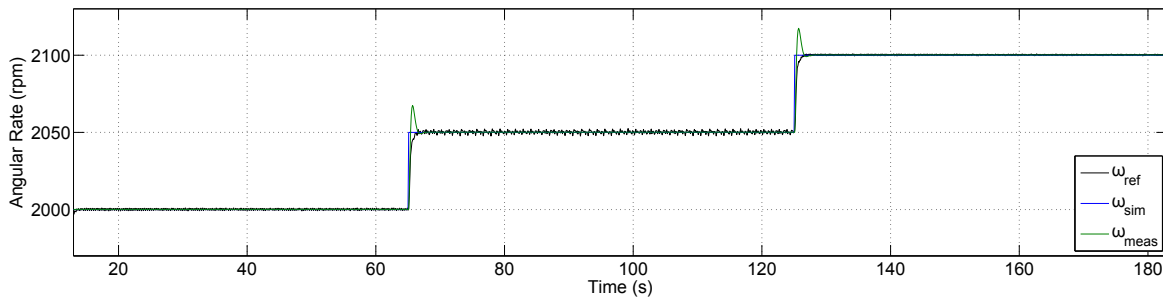
Figure 4.11 – The steady state RW test ($\omega_{bias} = 2000$ rpm)

$\overline{\omega}_{error}$ gives a more accurate representation of the RW's steady state control accuracy than σ_{meas}. It should also be noted that the measurements at $\omega_{ref} = 2050$ rpm were less accurate than the other sets of measurements. One possible explanation for the above-mentioned is that the resonant frequency of the RW might be close to 2050 rpm.

Figure 4.12 shows the ramp response of the RW as well as the rate error for the third test. The mean offset that can be seen in Figure 4.12 verifies the earlier assumption that the approximated plant integrator is not a pure integrator. The closed loop system was thus not able to follow a ramp input with a zero mean error.

ω_{ref}	ω_{mean}	ω_{offset}	σ_{meas}	$\overline{\omega}_{error}$
rpm	rpm	rpm	rpm	rpm
1000	1000.236	0.236	0.534	0.437
1050	1050.235	0.235	0.745	0.639
1100	1100.238	0.238	0.726	0.601
2000	2000.216	0.216	0.614	0.608
2050	2050.238	0.238	1.005	0.813
2100	2100.236	0.236	0.403	0.389

Table 4.2 – An analysis of the RW's steady state control accuracy
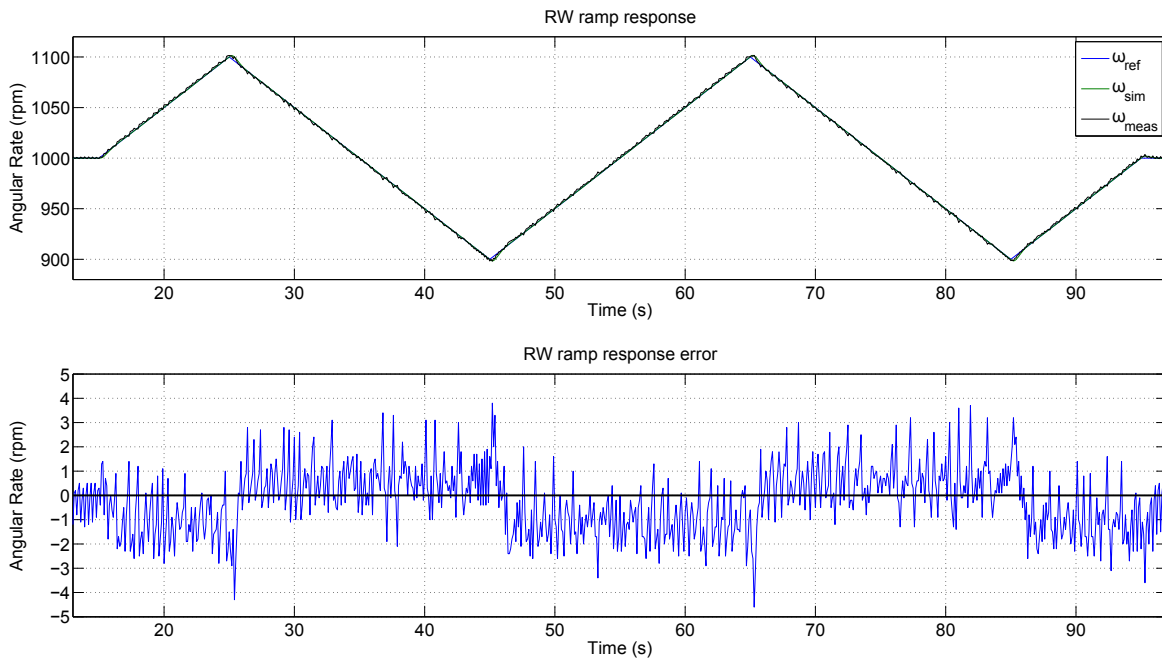
Figure 4.12 – The ramp response RW test

The Simulink model of the above-mentioned PID-controlled RW can be seen in Figure 4.13. The PWM limit feedback is implemented to ensure that the saturated PWM register value (and not the pre-saturation value) is reused by the controller (i.e. for the first term of Equation 4.5.6). The poles of the discrete controller in Figure 4.13 have also been adjusted accordingly. The input and outputs of the model was lastly scaled with the wheel moment of inertia I_{ww} and the appropriate conversion factor between rad/s and rpm.

Although the engineering model RW served as a guideline for creating the simulation environment RW model, the model was slightly altered before implementation. A commercial RW will most likely perform significantly better than the engineering model RW, since the alignment of the magnetic encoder on the engineering model was done by hand (which almost certainly implies an increase in measurement noise). The 1-σ measurement noise added to the wheel speed sensor in the simulation environment was thus set to 0.5 rpm (roughly 75% of the engineering model's noise), which implies a
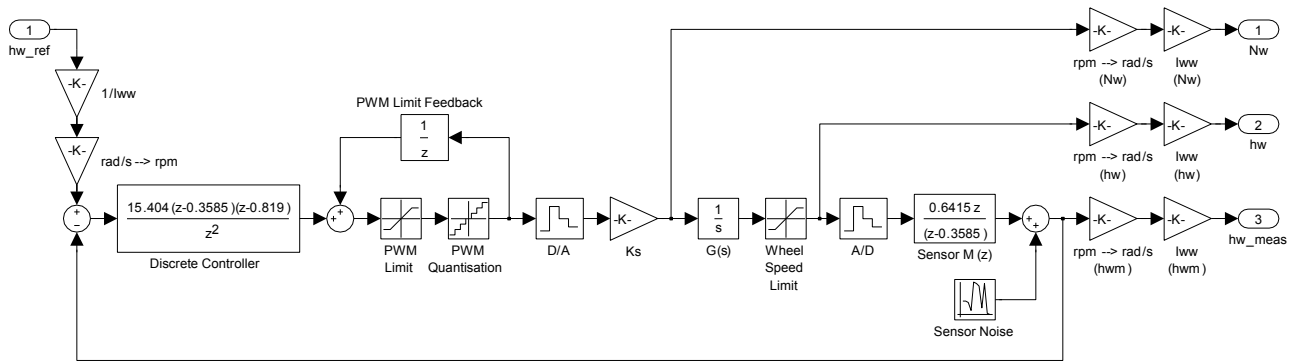
Figure 4.13 – The RW model in the simulation environment

maximum error of ± 1.5 rpm. The PWM quantisation was based on 12-bit accuracy as opposed to the 10-bit accuracy of the engineering model wheel. Lastly, the static and dynamic imbalances of the RW model are based on that of the RW-0.06 from Sinclair Interplanetary, which was found to be a suitable reaction wheel for this project during the initial ADCS analysis (see Chapter 3). The above-mentioned values are given as 0.0208 gcm and 0.0208 gcm^2 respectively.

4.6   Summary

This chapter summarised the development of an application specific ADCS simulation environment. The orbit of the satellite was firstly chosen, after which the positions of the imaging payload, communication payload, and main solar panel were defined for tracking purposes.

The derivation of the hardware models that were used in the ADCS simulation environment was also discussed in this chapter. The sensors' measured vectors were modelled based on certain reference vectors and the sensor noise was furthermore approximated as zero-mean Gaussian noise. The placement of the fine sun sensor and the star tracker on the satellite body was also discussed.

The actuator models were lastly defined. Although the torquers were assumed to only be affected by digital quantisation, a detailed reaction wheel model was derived with the use of a practical example.

Chapter 5

Attitude Determination

5.1 Overview

The determination of a satellite's current attitude plays a vital role in the process of achieving the desired mission goals. Before any controlled manoeuvre can be performed, the satellite's attitude control system requires knowledge of the current attitude. There is an enormous amount of attitude estimators in existence, ranging from simple geometric methods to complex non-linear methods. Various sensors can be used to feed these estimators.

This chapter will explore only a few of the attitude determination methods used on-board satellites. The estimators that will be discussed include the TRIAD and Optimized TRIAD methods, a Rate Kalman Filter (RKF) and a full state Extended Kalman Filter (EKF). The derivation and implementation of each estimator will also be shown and discussed.

Simulation results will be used to verify the working of each estimator and to compare the accuracies of the estimators. The effect of moment of inertia uncertainty on the performance of the EKF will also be discussed. The results that will be shown in this chapter are based on the orbit parameters and ADCS hardware specifications from Chapter 4, unless otherwise stated.

5.2 TRIAD and Optimized TRIAD

The TRIAD attitude determination algorithm is a geometric approach to estimating the satellite's attitude quaternion. TRIAD involves using two measured vectors (in SBC) and their respective modelled reference vectors (in ORC) to calculate the DCM that represents the transformation from the ORC frame to the SBC frame. The algorithm was first introduced by Harold Black in 1964 [17]. The Optimized TRIAD algorithm, "which provides an improved solution" compared to the original algorithm, was developed more than three decades later [18].

5.2.1   TRIAD Implementation

The first step of the TRIAD algorithm is creating two triads. A triad in this case refers to set of three orthogonal unit vectors. If the two measured vectors are $\mathbf{v}_1$ and $\mathbf{v}_2$, then the first triad, or $\mathbf{m}_1$, $\mathbf{m}_2$ and $\mathbf{m}_3$, is calculated as

$$\mathbf{m}_1 \;=\; \frac{\mathbf{v}_1}{||\mathbf{v}_1||} \;, \tag{5.2.1}$$

$$\mathbf{m}_2 \;=\; \frac{\mathbf{v}_1 \times \mathbf{v}_2}{||\mathbf{v}_1 \times \mathbf{v}_2||} \;, \tag{5.2.2}$$

$$\text{and} \quad \mathbf{m}_3 \;=\; \mathbf{m}_1 \times \mathbf{m}_2 \;. \tag{5.2.3}$$

It is suggested that the anchor vector, which is defined as the vector used to calculate $\mathbf{m}_1$ (in this case $\mathbf{v}_1$), should be the most accurate measured vector [18]. This process will maximise the estimation accuracy. The two reference vectors, or $\mathbf{w}_1$ and $\mathbf{w}_2$, are similarly used to create the second triad, or $\mathbf{r}_1$, $\mathbf{r}_2$ and $\mathbf{r}_3$, thus

$$\mathbf{r}_1 \;=\; \frac{\mathbf{w}_1}{||\mathbf{w}_1||} \;, \tag{5.2.4}$$

$$\mathbf{r}_2 \;=\; \frac{\mathbf{w}_1 \times \mathbf{w}_2}{||\mathbf{w}_1 \times \mathbf{w}_2||} \;, \tag{5.2.5}$$

$$\text{and} \quad \mathbf{r}_3 \;=\; \mathbf{r}_1 \times \mathbf{r}_2 \;. \tag{5.2.6}$$

It should also be noted that $\mathbf{w}_i$ is the reference vector associated with $\mathbf{v}_i$. The second step of TRIAD is to calculate the estimated attitude quaternion $\hat{\mathbf{q}}$. The estimated DCM $\mathbf{A}_1$ can be calculated using

$$\mathbf{A}_1 \;=\; \mathbf{m}_1\mathbf{r}_1^T + \mathbf{m}_1\mathbf{r}_1^T + \mathbf{m}_1\mathbf{r}_1^T \;, \tag{5.2.7}$$

from which $\hat{\mathbf{q}}$ can be determined as described in Section 1.3.

5.2.2   Optimized TRIAD Implementation

The Optimized TRIAD algorithm increases the accuracy of $\hat{\mathbf{q}}$ by calculating a second DCM, $\mathbf{A}_2$, and taking the weighted average of $\mathbf{A}_1$ and $\mathbf{A}_2$. The second DCM is calculated by repeating Equations 5.2.1 to 5.2.7, but using the least accurate measured vector as the anchor vector. The DCM resulting from the above-mentioned weighted average process, or $\hat{\mathbf{A}}'$, is not orthogonal and thus "not a legitimate attitude matrix" [18]. $\hat{\mathbf{A}}'$ must therefore be orthogonalised to create $\hat{\mathbf{A}}$.

If σ_1 and σ_2 are the respective standard deviations associated with the measured vectors $\mathbf{v}_1$ and $\mathbf{v}_2$, then $\hat{\mathbf{A}}'$ is calculated as

$$\hat{\mathbf{A}}' \;=\; \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}\mathbf{A}_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}\mathbf{A}_2 \;. \tag{5.2.8}$$

The orthogonalisation process can be expressed as

$$\hat{\mathbf{A}} \;=\; \frac{1}{2}\left[\hat{\mathbf{A}}' + (\hat{\mathbf{A}}'^{-1})^T\right] \;,$$ (5.2.9)

after which the estimated attitude quaternion $\hat{\mathbf{q}}$ can be determined.

5.2.3   TRIAD and Optimized TRIAD Simulation Results

Geometric attitude determination methods like TRIAD and Optimized TRIAD are non-recursive, which means that they provide solutions instantaneously after being activated. The choice of satellite state (eg. tumbling or stable) under which to verify the working of TRIAD and Optimized TRIAD is thus trivial. A comparison between the two algorithms was produced by running three identical simulations on an uncontrolled tumbling satellite state, after which accuracy of the Optimized TRIAD (which proved to be the more accurate estimator, as expected) was determined.

For the results shown in this section, the selected estimator was activated after 200 s. The initial angular rates $\boldsymbol{\omega}_B^I$ and Euler angles **RPY** were arbitrarily set to $\boldsymbol{\omega}_B^I = \begin{bmatrix} 2 & -1 & 4 \end{bmatrix}^T °/\text{s}$ and $\mathbf{RPY} = \begin{bmatrix} -20 & 30 & 10 \end{bmatrix}^T °$. A magnetometer and a CSS were used as the measured vectors. The modeled geomagnetic field vector $\mathbf{B}_o$, as generated by the IGRF model, also included low frequency noise of 1 μT_{RMS}.

Two simulations were performed with TRIAD, the first using the magnetometer measurement $\mathbf{B}_b$ as anchor vector and the second using the CSS measurement $\mathbf{S}_b$ as anchor vector. A third simulation was then performed with Optimized TRIAD. Figure 5.1 shows the root-mean-square (RMS) Euler angle errors of the three simulations. A clearer performance comparison between TRIAD and Optimized TRIAD is provided by the application of a low-pass filter on the estimated Euler angle errors.
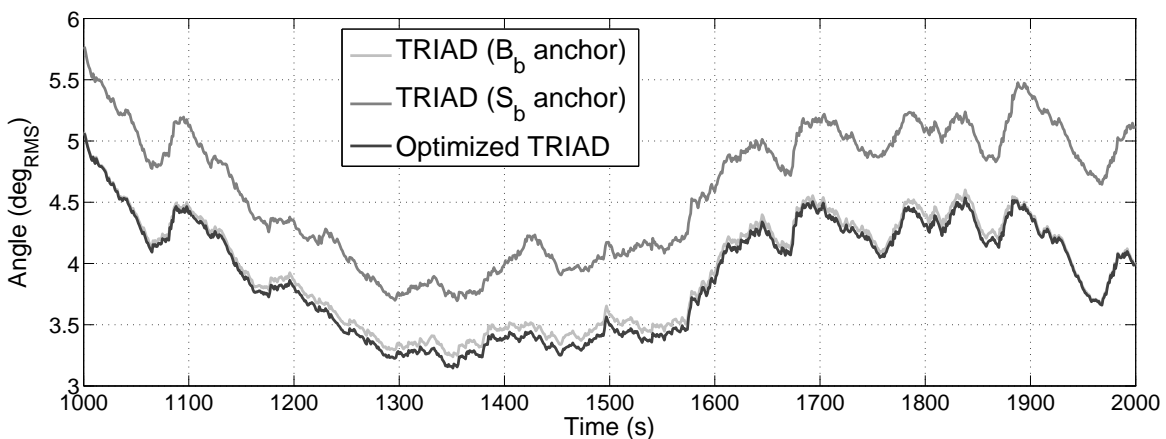
Figure 5.1 – A comparison between the TRIAD and Optimized TRIAD algorithms

From Figure 5.1 it can be concluded that Optimized TRIAD produced attitude estimates that are slightly more accurate than that of TRIAD. Figures 5.2(a) to 5.2(c) show the true and estimated

Euler angles of Optimized TRIAD. The standard deviations of the Euler angle errors in Figure 5.2(d) remained below 3.5°.

(a) Roll angle   (b) Pitch angle   (c) Yaw angle

(d) Optimized TRIAD angle errors
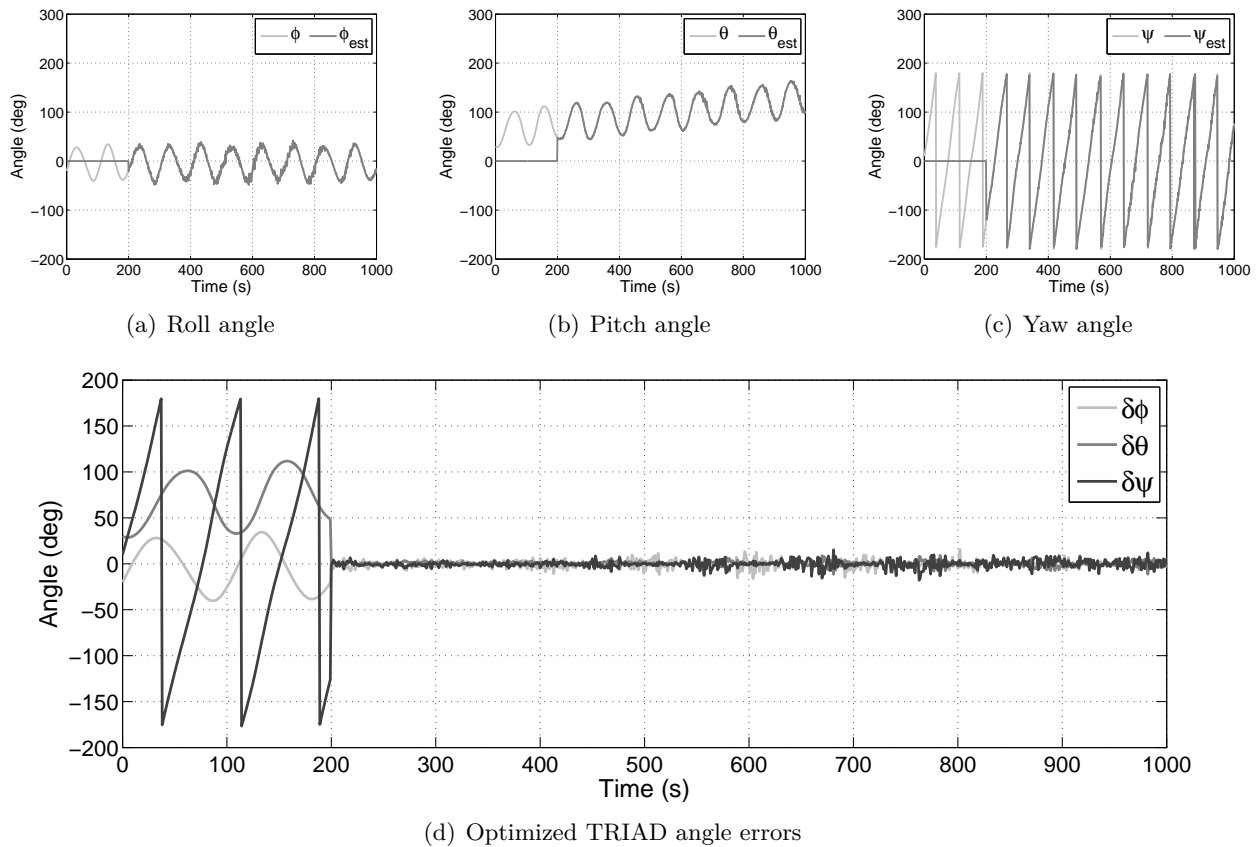
Figure 5.2 – The Optimized TRIAD performance during an uncontrolled tumble

5.3 Rate Kalman Filter

The Kalman Filter (KF), also known as the Kalman-Bucy Filter, is named after Rudolf E. Kalman, who is accredited with co-inventing this optimal linear estimator in the early 1960s [19; 20]. The KF is particularly useful when trying to estimate states that cannot be measured in the presence of noisy measurements, as is mostly the case with a satellite's ADCS.

The RKF uses a 3-axis vector measurement to estimate the body angular rates of the satellite. Although the RKF was first developed to be used with a magnetometer during de-tumbling [16], a modified sun-sensor-based version thereof can also be useful during normal operation of the satellite if fine pointing is not required [11].

As with the KF, the RKF consists of a linear system model and measurement model. The system model represents the internal dynamics of the system being estimated, while the measurement model reflects the relationship between the measured vector and the estimated state vector. For a continuous

system the general form of the system model is

$$\dot{\mathbf{x}}_t \;\; = \;\; \mathbf{F}_t\mathbf{x}_t + \mathbf{G}_t\mathbf{u}_t + \mathbf{s}_t \tag{5.3.1}$$

and the general form of the measurement model is

$$\mathbf{y}_t \;\; = \;\; \mathbf{H}_t\mathbf{x}_t + \mathbf{m}_t \; , \tag{5.3.2}$$

where $\mathbf{x}_t$ is the state vector, $\mathbf{u}_t$ is the system input, $\mathbf{s}_t$ is the system noise, $\mathbf{y}_t$ is the measurement output and $\mathbf{m}_t$ is the measurement noise.

5.3.1 RKF System Model

The satellite dynamics is the system being estimated and it is represented by the well-known Euler dynamic equation shown in Equation 1.3.24. For the RKF the aforementioned equation can be linearised and rewritten as

$$\dot{\boldsymbol{\omega}}_B^I \;\; = \;\; \mathbf{J}^{-1}(\mathbf{N}_m - \mathbf{N}_w + \mathbf{N}_d) \; . \tag{5.3.3}$$

$\mathbf{N}_{gyro}$ from Equation 1.3.24 originates from a non-linear function of $\boldsymbol{\omega}_B^I$ is thus included in the system noise $\mathbf{N}_d$. $\mathbf{N}_m$ and $\mathbf{N}_w$ on the other hand are known control torques.

If $\boldsymbol{\omega}_B^I$ is represented by a state vector $\mathbf{x}$ (as opposed to $\boldsymbol{\omega}_B^O$ in the original RKF [16]), then the RKF continuous system model can be written as

$$\dot{\mathbf{x}}_t \;\; = \;\; \mathbf{F}_t\mathbf{x}_t + \mathbf{G}_t\mathbf{u}_t + \mathbf{s}_t \; , \tag{5.3.4}$$

$$\text{where} \quad \mathbf{F}_t \;\; = \;\; \mathbf{0} \; ,$$
$$\mathbf{G}_t \;\; = \;\; \mathbf{J}^{-1} \; ,$$
$$\mathbf{u}_t \;\; = \;\; \mathbf{N}_{m,t} - \mathbf{N}_{w,t} \; ,$$
$$\text{and} \quad \mathbf{s}_t \;\; = \;\; \mathbf{J}^{-1}\mathbf{N}_{d,t} \; .$$

It is assumed that the system noise $\mathbf{s}_t$ is zero-mean with covariance matrix $\mathbf{Q}_t$. The discrete domain equivalent of the RKF system model is determined from the continuous model as

$$\mathbf{x}_{k+1} \;\; = \;\; \boldsymbol{\Phi}_k\mathbf{x}_k + \boldsymbol{\Gamma}_k\mathbf{u}_k + \mathbf{s}_k \; , \tag{5.3.5}$$

$$\text{where} \quad \boldsymbol{\Phi}_k \;\; = \;\; e^{T_s[\mathbf{F}_t]_{t=kT_s}} \;\; = \;\; \mathbf{I}_{3\times3} \; ,$$
$$\boldsymbol{\Gamma}_k \;\; \approx \;\; T_s \left[\mathbf{G}_t\right]_{t=kT_s} \; ,$$
$$\mathbf{u}_k \;\; = \;\; [\mathbf{N}_{M,t} - \mathbf{N}_{W,t}]_{t=kT_s} \; ,$$
$$\text{and} \quad \mathbf{s}_k \;\; = \;\; [\mathbf{s}_t]_{t=kT_s} \; ,$$

where T_s is the sample time and the discrete system noise $\mathbf{s}_k$, represented by the covariance matrix $\mathbf{Q}_k$, is assumed to be zero-mean and Gaussian.

5.3.2 RKF Measurement Model

The RKF measurement model relies strongly on the assumption that $\boldsymbol{\omega}_B^O \approx \boldsymbol{\omega}_B^I$ and that $\boldsymbol{\omega}_B^O$ is near-constant between successive measurements (with a sampling period of T_s). The change in the Euler angles over a period T_s at sampling instance k, given by $(\Delta\phi)_k$, $(\Delta\theta)_k$ and $(\Delta\psi)_k$, can thus be approximated as

$$\begin{bmatrix} (\Delta\phi)_k & (\Delta\theta)_k & (\Delta\psi)_k \end{bmatrix}^T \approx \begin{bmatrix} T_s(\omega_{ix})_k & T_s(\omega_{iy})_k & T_s(\omega_{iz})_k \end{bmatrix}^T .$$

Using the small-angle approximation from Section 1.3, the transformation matrix $(\Delta\mathbf{A})_k$ that represents the change in attitude between sampling instances $k-1$ and k can be expressed as

$$
\begin{aligned}
(\Delta\mathbf{A})_k &\approx \begin{bmatrix} 1 & (\Delta\psi)_k & -(\Delta\theta)_k \\ -(\Delta\psi)_k & 1 & (\Delta\phi)_k \\ (\Delta\theta)_k & -(\Delta\phi)_k & 1 \end{bmatrix} \\[2mm]
&\approx \begin{bmatrix} 1 & T_s(\omega_{iz})_k & -T_s(\omega_{iy})_k \\ -T_s(\omega_{iz})_k & 1 & T_s(\omega_{ix})_k \\ T_s(\omega_{iy})_k & -T_s(\omega_{ix})_k & 1 \end{bmatrix}
\end{aligned} \tag{5.3.6}
$$

Since $\Delta\mathbf{A}$ represents the change in attitude, it also represents the change in unit measurement vectors between successive sampling instances. If $\mathbf{v}_{k-1}$ and $\mathbf{v}_k$ are successive unit measurement vectors, then

$$\mathbf{v}_k = (\Delta\mathbf{A})_k\, \mathbf{v}_{k-1} . \tag{5.3.7}$$

The difference between two successive measurements $(\Delta\mathbf{v})_k$ can thus be expressed using Equation 5.3.7 as

$$
\begin{aligned}
(\Delta\mathbf{v})_k &= \mathbf{v}_k - \mathbf{v}_{k-1} \tag{5.3.8} \\[2mm]
&= (\Delta\mathbf{A})_k\, \mathbf{v}_{k-1} - \mathbf{v}_{k-1} \\[2mm]
&= \left[(\Delta\mathbf{A})_k - \mathbf{I} \right] \mathbf{v}_{k-1} \\[2mm]
&= \begin{bmatrix} 0 & T_s(\omega_{iz})_k & -T_s(\omega_{iy})_k \\ -T_s(\omega_{iz})_k & 0 & T_s(\omega_{ix})_k \\ T_s(\omega_{iy})_k & -T_s(\omega_{ix})_k & 0 \end{bmatrix} \mathbf{v}_{k-1} .
\end{aligned} \tag{5.3.9}
$$

Inspection reveals that the matrix-vector-multiplication from Equation 5.3.9 can be rewritten in terms of $(\boldsymbol{\omega}_B^I)_k$. The measurement model of the RKF can now be expressed in the general form of Equation

5.3.2 as

$$
\begin{aligned}
\mathbf{y}_k &= (\Delta\mathbf{v})_k \\[2mm]
&= \begin{bmatrix} 0 & T_s(v_z)_{k-1} & -T_s(v_y)_{k-1} \\[2mm] -T_s(v_z)_{k-1} & 0 & T_s(v_x)_{k-1} \\[2mm] T_s(v_y)_{k-1} & -T_s(v_x)_{k-1} & 0 \end{bmatrix} \begin{bmatrix} (\omega_{ix})_k \\[2mm] (\omega_{iy})_k \\[2mm] (\omega_{iz})_k \end{bmatrix} + \mathbf{m}_k \\[2mm]
&= \mathbf{H}_k(\boldsymbol{\omega}_B^I)_k + \mathbf{m}_k \ .
\end{aligned}
\tag{5.3.10}
$$

The discrete covariance matrix of the measurement noise $\mathbf{m}_k$, which is assumed to be zero-mean and Gaussian, is $\mathbf{R}_k$.

5.3.3   RKF Implementation

The implementation of the RKF consists of a prediction step (or model update) and a correction step (or measurement update). During the model update the estimated state vector $\hat{\mathbf{x}}_k$ and state covariance matrix $\mathbf{P}_k$ are predicted (or propagated) based on the system model. The measurement update is then performed to correct $\hat{\mathbf{x}}_k$ and $\mathbf{P}_k$ based on the measurement vector. To avoid confusion between the model update and the measurement update, a '$-$' superscript will be used to indicate a predicted vector or matrix (i.e. from the model update). Similarly, a '$+$' superscript will be used to indicate that the given vector or matrix has been corrected by the measurement update.

Model Update

Since the state vector $\hat{\mathbf{x}}_k$ is the inertial-referenced angular rate $\hat{\boldsymbol{\omega}}_B^I$ for the RKF, $\hat{\mathbf{x}}_k$ can be propagated by using Equation 5.3.3 and a modified Euler integration. The propagation of $\hat{\mathbf{x}}_k$ can thus be performed using

$$
(d\hat{\boldsymbol{\omega}}_B^I)_k = \mathbf{J}^{-1}\Big((\mathbf{N}_m)_{k-1} - (\mathbf{N}_w)_{k-1} - (\mathbf{N}_{gyro})_{k-1}\Big) \tag{5.3.11}
$$

$$
\text{and} \quad \hat{\mathbf{x}}_k^- = \hat{\mathbf{x}}_{k-1}^+ + \frac{T_s}{2}\Big(3(d\hat{\boldsymbol{\omega}}_B^I)_k - (d\hat{\boldsymbol{\omega}}_B^I)_{k-1}\Big) \ . \tag{5.3.12}
$$

The Euler integration of Equation 5.3.11 will result in a small error, but the error is insignificant with a sample period of 1 second. The propagation of $\mathbf{P}_k$ during the model update is executed via

$$
\mathbf{P}_k^- = \Phi_k \mathbf{P}_{k-1}^+ \Phi_k^T + \mathbf{Q}_k \ . \tag{5.3.13}
$$

Measurement Update

The measurement update should now be performed to correct the estimated state vector and the state covariance matrix using

$$\hat{\mathbf{x}}_k^+ \;=\; \hat{\mathbf{x}}_k^- + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{H}_k\hat{\mathbf{x}}_k^-\right) \tag{5.3.14}$$

$$\text{and}\quad \mathbf{P}_k^+ \;=\; \left[\mathbf{I}_{3\times3} - \mathbf{K}_k\mathbf{H}_k\right]\mathbf{P}_k^- , \tag{5.3.15}$$

$$\text{where}\quad \mathbf{K}_k \;=\; \mathbf{P}_k^-\mathbf{H}_k^T\left[\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k\right]^{-1} . \tag{5.3.16}$$

5.3.4   RKF Simulation Results

The performance of the RKF was investigated by simulating two satellite states: (1) a tumbling state and (2) a stable state. The RKF can firstly be used in the process of de-tumbling the satellite, i.e. the tumbling state. The RKF can furthermore be used when the satellite is stable but does not require highly accurate rate estimates, i.e. the stable state.

For the **tumbling state** (e.g. when the satellite is released), the initial body angular rates are expected to be less than $10°/\text{s}$ [15] and were thus chosen as $\boldsymbol{\omega}_B^I = \begin{bmatrix} 2 & -1 & 4 \end{bmatrix}^T °/\text{s}$. The initial Euler angles are trivial when de-tumbling the satellite and were thus arbitrarily chosen as $\mathbf{RPY} = \begin{bmatrix} -20 & 30 & 10 \end{bmatrix}^T °$. No actuators were active during the simulation, thus $\mathbf{N}_m = \mathbf{N}_w = \mathbf{h}_w = \mathbf{0}$. A magnetometer-based RKF was activated after 200 s during the simulation of the tumbling state.

The angular rates during the **stable state** are typically very low. For the results pertaining to the stable state, the RKF was based on fine sun sensor measurements, which meant that only the sunlit part of the orbit was relevant. Once again, no actuators were active during the simulation, thus $\mathbf{N}_m = \mathbf{N}_w = \mathbf{h}_w = \mathbf{0}$.

In both simulations a 1% moment of inertia error and a 1 μT_{RMS} IGRF noise were added.

Tumbling State

Figures 5.3(a) to 5.3(c) show the true and estimated angular rates and Figure 5.3(d) shows estimation errors of the RKF during the tumbling state. The RMS angular rate error stabilises after 200 seconds and remains below $\pm 0.3°/\text{s}$.

Stable State

Figures 5.4(a) to 5.4(c) show the true and estimated angular rates and Figure 5.4(d) shows estimation errors during the stable state, where the angular rate errors can be seen to remain within $\pm 0.015°/\text{s}$. The large initial Y-rate error is a result of the approximation of $\boldsymbol{\omega}_B^O \approx \boldsymbol{\omega}_B^I$.

(a) Body X-axis rate

(b) Body Y-axis rate

(c) Body Z-axis rate

(d) RKF Angular Rate Errors
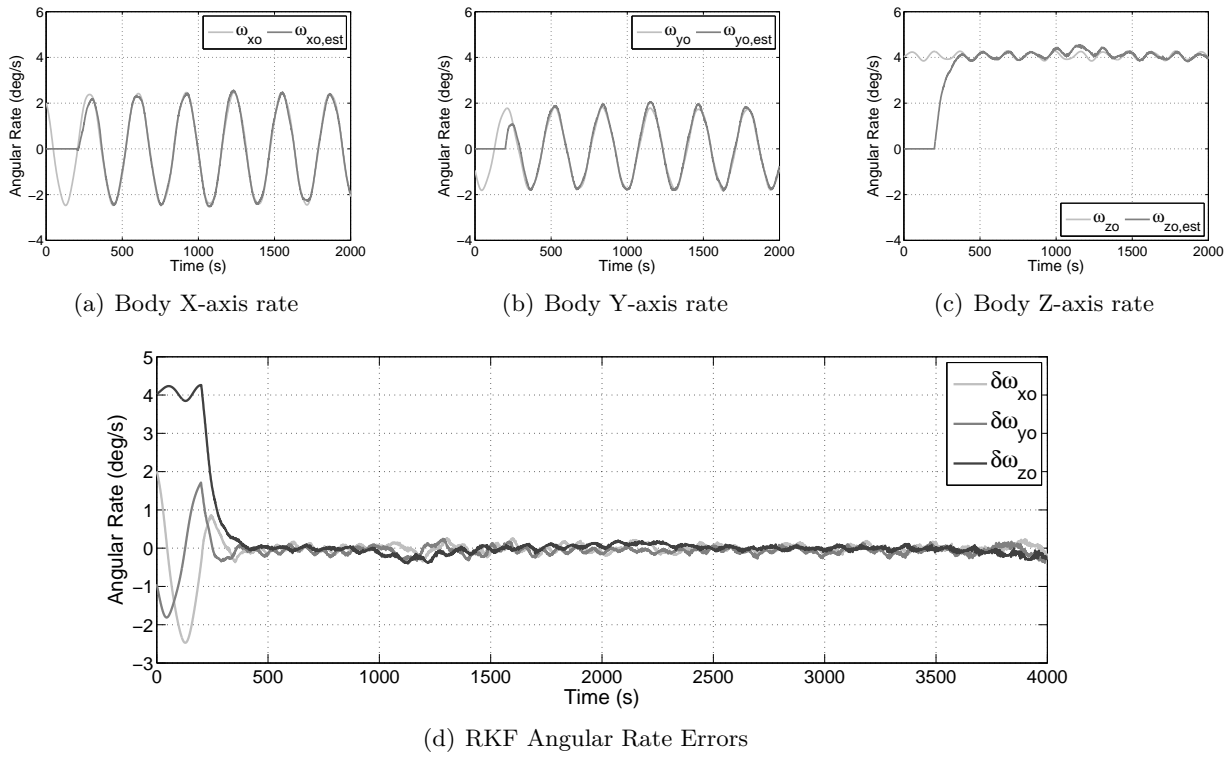
Figure 5.3 – The RKF performance in a tumbling state

(a) Body X-axis rate

(b) Body Y-axis rate

(c) Body Z-axis rate

(d) RKF Angular Rate Errors
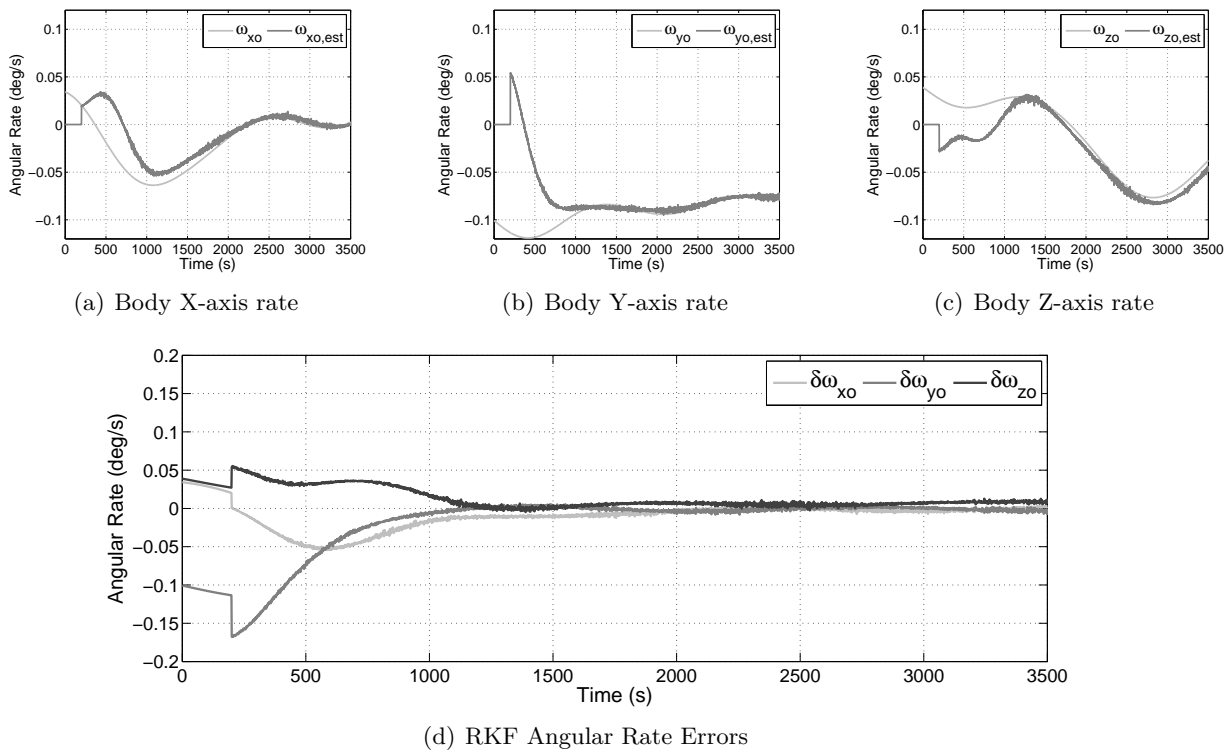
Figure 5.4 – The RKF performance in a stable state

5.4 Full State Extended Kalman Filter

The EKF is a variation of the KF that is extremely useful when dealing with non-linear systems. The EKF also consists of a system model and a measurement model, of which at least one is non-linear. For a continuous system the general form of the system model is

$$\dot{\mathbf{x}}_t \;=\; \boldsymbol{f}(\mathbf{x}_t) + \mathbf{s}_t \tag{5.4.1}$$

and the general form of the measurement model is

$$\mathbf{y}_t \;=\; \boldsymbol{h}(\mathbf{x}_t) + \mathbf{m}_t \;, \tag{5.4.2}$$

where $\mathbf{x}_t$ is the state vector, $\mathbf{s}_t$ is the system noise, $\mathbf{y}_t$ is the measurement output and $\mathbf{m}_t$ is the measurement noise. $\boldsymbol{f}(\mathbf{x}_t)$ and $\boldsymbol{h}(\mathbf{x}_t)$ are both non-linear functions of $\mathbf{x}_t$.

The chosen EKF for this project is a full 7-state EKF [21] that uses a 3-axis measurement vector and its corresponding reference vector to estimate the satellite's attitude. The state vector $\mathbf{x}$ of the full state EKF, or

$$\mathbf{x} \;=\; \begin{bmatrix} \boldsymbol{\omega}_B^I & \mathbf{q} \end{bmatrix}^T \;, \tag{5.4.3}$$

fully describes the satellite's current attitude and angular rates.

5.4.1 EKF System Perturbation Model

Given the non-linear nature of the relationship between $\boldsymbol{\omega}_B^I$ and $\mathbf{q}$, a process of linearisation is required. The first step in this process is to approximate the non-linear function $\boldsymbol{f}(\mathbf{x}_t)$ from Equation 5.4.1 by using a Taylor series expansion. If $\mathbf{x}_t$ is the true state vector and $\hat{\mathbf{x}}_t$ is the estimated state vector, then

$$\begin{aligned}
\boldsymbol{f}(\mathbf{x}_t) &\;=\; \boldsymbol{f}(\hat{\mathbf{x}}_t) + \left[\frac{\partial \boldsymbol{f}}{\partial \mathbf{x}_t}\right]_{\mathbf{x}_t=\hat{\mathbf{x}}_t}(\mathbf{x}_t - \hat{\mathbf{x}}_t) + \frac{1}{2!}\left[\frac{\partial^2 \boldsymbol{f}}{\partial \mathbf{x}_t^2}\right]_{\mathbf{x}_t=\hat{\mathbf{x}}_t}(\mathbf{x}_t - \hat{\mathbf{x}}_t)^2 + ... \\
&\;\approx\; \boldsymbol{f}(\hat{\mathbf{x}}_t) + \mathbf{F}_t \Delta\mathbf{x}_t \;,
\end{aligned} \tag{5.4.4}$$

$$\text{where} \quad \mathbf{F}_t \;=\; \left[\frac{\partial \boldsymbol{f}}{\partial \mathbf{x}_t}\right]_{\mathbf{x}_t=\hat{\mathbf{x}}_t} \tag{5.4.5}$$

$$\text{and} \quad \Delta\mathbf{x}_t \;=\; \mathbf{x}_t - \hat{\mathbf{x}}_t \;. \tag{5.4.6}$$

By taking the time derivative of Equation 5.4.6 and manipulating Equation 5.4.4, Equation 5.4.6 becomes

$$\begin{aligned}
\Delta\dot{\mathbf{x}}_t &\;=\; \dot{\mathbf{x}}_t - \dot{\hat{\mathbf{x}}}_t \\
&\;=\; \left(\boldsymbol{f}(\mathbf{x}_t) + \mathbf{s}_t\right) - \left(\boldsymbol{f}(\hat{\mathbf{x}}_t)\right) \\
&\;=\; \left(\boldsymbol{f}(\mathbf{x}_t) - \boldsymbol{f}(\hat{\mathbf{x}}_t)\right) + \mathbf{s}_t \\
&\;=\; \mathbf{F}_t \Delta\mathbf{x}_t + \mathbf{s}_t \;.
\end{aligned} \tag{5.4.7}$$

Equation 5.4.7 is the linearised continuous system perturbation model. A full derivation of the Jacobian matrix $\mathbf{F}_t$ can be found in Appendix B. The discrete equivalent of $\mathbf{F}_t$, namely $\boldsymbol{\Phi}_k$, can be calculated via

$$\boldsymbol{\Phi}_k \;\; = \;\; \left[e^{T_s \mathbf{F}_t} \right]_{\mathbf{x}=\hat{\mathbf{x}},\ t=kT_s} \; . \tag{5.4.8}$$

The expansion of the exponential in Equation 5.4.8 converges to a satisfactory extent after three terms [21], which means that it can be simplified as

$$\boldsymbol{\Phi}_k \;\; \approx \;\; \left[\mathbf{I} + T_s \mathbf{F}_t + \tfrac{1}{2} T_s{}^2 \mathbf{F}_t{}^2 \right]_{\mathbf{x}=\hat{\mathbf{x}},\ t=kT_s} \; . \tag{5.4.9}$$

The discrete system perturbation model can now be expressed in terms of $\boldsymbol{\Phi}_k$ as

$$\Delta\mathbf{x}_{k+1} \;\; = \;\; \boldsymbol{\Phi}_k \Delta\mathbf{x}_k \; . \tag{5.4.10}$$

5.4.2 EKF Measurement Perturbation Model

A process of linearisation is also required to determine the EKF measurement perturbation model. The Taylor series expansion from Equation 5.4.4 should once again be used to approximate the non-linear function $\boldsymbol{h}(\mathbf{x}_t)$. Since the continuous and discrete measurement perturbation models are equal, the Taylor series expansion results in

$$\boldsymbol{h}(\mathbf{x}_k) \;\; \approx \;\; \boldsymbol{h}(\hat{\mathbf{x}}_k) + \mathbf{H}_k \Delta\mathbf{x}_k \; , \tag{5.4.11}$$

$$\text{where} \quad \mathbf{H}_k \;\; = \;\; \left[\tfrac{\partial \boldsymbol{h}}{\partial \mathbf{x}_k} \right]_{\mathbf{x}_k = \hat{\mathbf{x}}_k} \; . \tag{5.4.12}$$

The relationship between the true (or noiseless) measured vector $\overline{\mathbf{v}}_{meas,k}$ (in SBC) and the true modelled vector $\overline{\mathbf{v}}_{model,k}$ (in ORC) at time step k can furthermore be expressed as a DCM rotation, thus

$$\overline{\mathbf{v}}_{meas,k} \;\; = \;\; \mathbf{A}_k \overline{\mathbf{v}}_{model,k} \; . \tag{5.4.13}$$

If the discrete measurement model output $\mathbf{y}_k$ is defined to be the measured vector $\mathbf{v}_{meas,k}$, then the non-linear output model can be expressed as

$$\mathbf{y}_k \;\; = \;\; \mathbf{v}_{meas,k} \tag{5.4.14}$$

$$\;\; = \;\; \mathbf{A}_k \mathbf{v}_{model,k} + \mathbf{m}_k \; . \tag{5.4.15}$$

The estimated measurement model output $\hat{\mathbf{y}}_k$ can thus be expressed as a function of the estimated DCM $\hat{\mathbf{A}}_k$ as

$$\hat{\mathbf{y}}_k \;\; = \;\; \hat{\mathbf{A}}_k \mathbf{v}_{model,k} \; . \tag{5.4.16}$$

If the innovation (or the output prediction error) $\mathbf{e}_k$ is defined as

$$\mathbf{e}_k \quad = \quad \mathbf{y}_k - \hat{\mathbf{y}}_k \; , \tag{5.4.17}$$

then Equations 5.4.11 and 5.4.17 can be used to determine the discrete measurement perturbation model as

$$
\begin{aligned}
\mathbf{e}_k \quad &= \quad \mathbf{y}_k - \hat{\mathbf{y}}_k \\
&= \quad \Big(\boldsymbol{h}(\mathbf{x}_k) + \mathbf{m}_k \Big) - \Big(\boldsymbol{h}(\hat{\mathbf{x}}_k) \Big) \\
&= \quad \Big(\boldsymbol{h}(\mathbf{x}_k) - \boldsymbol{h}(\hat{\mathbf{x}}_k) \Big) + \mathbf{m}_k \\
&= \quad \mathbf{H}_k \Delta \mathbf{x}_k + \mathbf{m}_k \; .
\end{aligned}
\tag{5.4.18}
$$

$\mathbf{m}_k$ is the innovation noise, which is zero-mean Gaussian noise with covariance matrix $\mathbf{R}_k$. Full derivations of $\mathbf{H}_k$ and $\mathbf{R}_k$ can be found in Appendix B.

5.4.3 EKF Implementation

The implementation of the EKF, similar to that of the RKF, also consists of a prediction step (or model update) and a correction step (or measurement update). During the model update the estimated state vector $\hat{\mathbf{x}}_k$ and state covariance matrix $\mathbf{P}_k$ are predicted (or propagated) based on the system model. The measurement update is then performed to correct $\hat{\mathbf{x}}_k$ and $\mathbf{P}_k$ based on the measurement vector. The superscripts '−' and '+' will once again be used to respectively indicate a predicted vector or matrix and a corrected vector or matrix.

Model Update

The propagation of the state vector $\hat{\mathbf{x}}_k$ must be separated into the propagation of $(\hat{\boldsymbol{\omega}}_B^I)_k$ and the propagation of $\hat{\mathbf{q}}_k$. The propagation of $(\hat{\boldsymbol{\omega}}_B^I)_k$ can be performed by integrating the result of the Euler dynamic equation, or

$$(d\hat{\boldsymbol{\omega}}_B^I)_k \quad = \quad \mathbf{J}^{-1}\Big((\mathbf{N}_m)_{k-1} - (\mathbf{N}_w)_{k-1} - (\mathbf{N}_{gg})_{k-1} - (\mathbf{N}_{gyro})_{k-1} \Big) \; . \tag{5.4.19}$$

As with the RKF, a discrete numerical Euler integration technique will suffice, thus

$$(\hat{\boldsymbol{\omega}}_B^I)_k^- \quad = \quad (\hat{\boldsymbol{\omega}}_B^I)_{k-1}^+ + \frac{T_s}{2}\Big(3(d\hat{\boldsymbol{\omega}}_B^I)_k - (d\hat{\boldsymbol{\omega}}_B^I)_{k-1} \Big) \; . \tag{5.4.20}$$

The propagation of $\hat{\mathbf{q}}_k$ can be performed using

$$\hat{\mathbf{q}}_k^- = \left[\cos\left(k_q\right)\mathbf{I}_{4\times4} + \frac{1}{\|(\hat{\boldsymbol{\omega}}_B^O)_k^-\|}\sin\left(k_q\right)\boldsymbol{\Omega}_k^-\right]\hat{\mathbf{q}}_{k-1}^+ , \tag{5.4.21}$$

$$\text{where} \quad k_q = \frac{T_s}{2}\left\|(\hat{\boldsymbol{\omega}}_B^O)_k^-\right\| ,$$

$$(\hat{\boldsymbol{\omega}}_B^O)_k^- = (\hat{\boldsymbol{\omega}}_B^I)_k^- - \hat{\mathbf{A}}_k\begin{bmatrix}0 & -(\omega_o)_k & 0\end{bmatrix}^T$$

$$= \begin{bmatrix}\hat{\omega}_{ox} & \hat{\omega}_{oy} & \hat{\omega}_{oz}\end{bmatrix}^T ,$$

$$\left\|(\hat{\boldsymbol{\omega}}_B^O)_k^-\right\| = \sqrt{\hat{\omega}_{ox}{}^2 + \hat{\omega}_{oy}{}^2 + \hat{\omega}_{oz}{}^2} ,$$

$$\text{and} \quad \boldsymbol{\Omega}_k^- = \begin{bmatrix}0 & \hat{\omega}_{oz} & -\hat{\omega}_{oy} & \hat{\omega}_{ox} \\ -\hat{\omega}_{oz} & 0 & \hat{\omega}_{ox} & \hat{\omega}_{oy} \\ \hat{\omega}_{oy} & -\hat{\omega}_{ox} & 0 & \hat{\omega}_{oz} \\ -\hat{\omega}_{ox} & -\hat{\omega}_{oy} & -\hat{\omega}_{oz} & 0\end{bmatrix} .$$

Once $(\hat{\boldsymbol{\omega}}_B^I)_k$ and $\hat{\mathbf{q}}_k$ have been propagated, the propagated state vector can be expressed as

$$\hat{\mathbf{x}}_k^- = \begin{bmatrix}(\hat{\boldsymbol{\omega}}_B^I)_k^- & \hat{\mathbf{q}}_k^-\end{bmatrix}^T .$$

The state covariance matrix $\mathbf{P}_k$ can be propagated via Equation 5.4.22, or

$$\mathbf{P}_k^- = \Phi_k\mathbf{P}_{k-1}^+\Phi_k^T + \mathbf{Q}_k , \tag{5.4.22}$$

which concludes the model update.

Measurement Update

The first step of the EKF measurement update is calculating the gain matrix $\mathbf{K}_k$ as

$$\mathbf{K}_k = \mathbf{P}_k^-(\mathbf{H}_k^-)^T\left[\mathbf{H}_k^-\mathbf{P}_k^-(\mathbf{H}_k^-)^T + \mathbf{R}_k\right]^{-1} , \tag{5.4.23}$$

$$\text{where} \quad \mathbf{H}_k^- = \left[\mathbf{H}_k\right]_{\mathbf{x}_k=\hat{\mathbf{x}}_k^-}$$

The estimated state vector can now be corrected by

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k\mathbf{e}_k , \tag{5.4.24}$$

$$\text{where} \quad \mathbf{e}_k = \mathbf{v}_{meas,k} - \hat{\mathbf{A}}_k^-\mathbf{v}_{model,k} .$$

The addition in Equation 5.4.24 will however force the magnitude of $\hat{\mathbf{q}}_k^+$ to be more than 1. $\hat{\mathbf{q}}_k^+$ must therefore be normalised after correcting the estimated state vector to "preserve the quaternion norm constraint" [21].

The correction of the state covariance matrix can be performed using

$$\mathbf{P}_k^+ \;=\; \left[\mathbf{I}_{7\times7} - \mathbf{K}_k\mathbf{H}_k^+\right]\mathbf{P}_k^-\left[\mathbf{I}_{7\times7} - \mathbf{K}_k\mathbf{H}_k^+\right]^{-1} + \mathbf{K}_k\mathbf{R}_k\mathbf{K}_k^T \;, \tag{5.4.25}$$

where $\quad \mathbf{H}_k^+ \;=\; \left[\mathbf{H}_k\right]_{\mathbf{x}_k=\hat{\mathbf{x}}_k^+}\;,$

which concludes the measurement update of the EKF.

5.4.4 EKF Simulation Results

The performance of the EKF was investigated by simulating two satellite states: (1) an uncontrolled tumbling state and (2) a controlled stable state. Although the EKF will mainly be used when the satellite is stable or when it is performing a controlled manoeuvre (i.e. the controlled stable state), an uncontrolled tumbling state is included to test the versatility of the EKF. A comparison between the results of the RKF and that of the EKF can thus also be made.

The initial angular rates for the **uncontrolled tumbling state** were set to $\boldsymbol{\omega}_B^I = \begin{bmatrix} 2 & -1 & 4 \end{bmatrix}^T °/$s, as with the RKF simulation from Section 5.3.4. The initial Euler angles were once again chosen as $\mathbf{RPY} = \begin{bmatrix} -20 & 30 & 10 \end{bmatrix}^T °$. No actuators were active during this simulation, thus $\mathbf{N}_m = \mathbf{N}_w = \mathbf{h}_w = \mathbf{0}$. For the simulation results related to the uncontrolled tumbling state, a magnetometer-based EKF was activated after 200 s.

The performance of the EKF during the **controlled stable state** was determined by controlling the angular rates and Euler angles to certain desired values. The controllers were assumed to be ideal (i.e. true rates and quaternions, no actuator noise and no model uncertainties) to isolate the true rates and quaternions from the EKF. The EKF was however affected by disturbance torques and MoI uncertainty. For the results generated during the controlled stable state, the EKF was based on star tracker measurements and activated after 200 s.

Uncontrolled Tumbling State

Figure 5.5 shows the true and estimated Euler angles and angular rates of the EKF during an uncontrolled tumble. The estimation errors stabilise after approximately 1000 seconds and remain within $\pm18°$ and $\pm0.13°/$s respectively.

The estimation errors in Figures 5.5(a) to 5.5(f) are however larger than expected. Investigation showed that the IGRF errors (see Section 2.2.2) caused the largest Euler angle errors, while the MoI error caused the largest angular rate errors. Without these uncertainties, the Euler angle errors and the angular rate errors remain within $\pm7°$ and $\pm0.05°/$s respectively. Figure 5.6 shows the RMS angular rate errors with and without the MoI error, confirming the result of the above-mentioned investigation.

(a) Roll angle   (b) Pitch angle   (c) Yaw angle

(d) Body X-axis rate   (e) Body Y-axis rate   (f) Body Z-axis rate
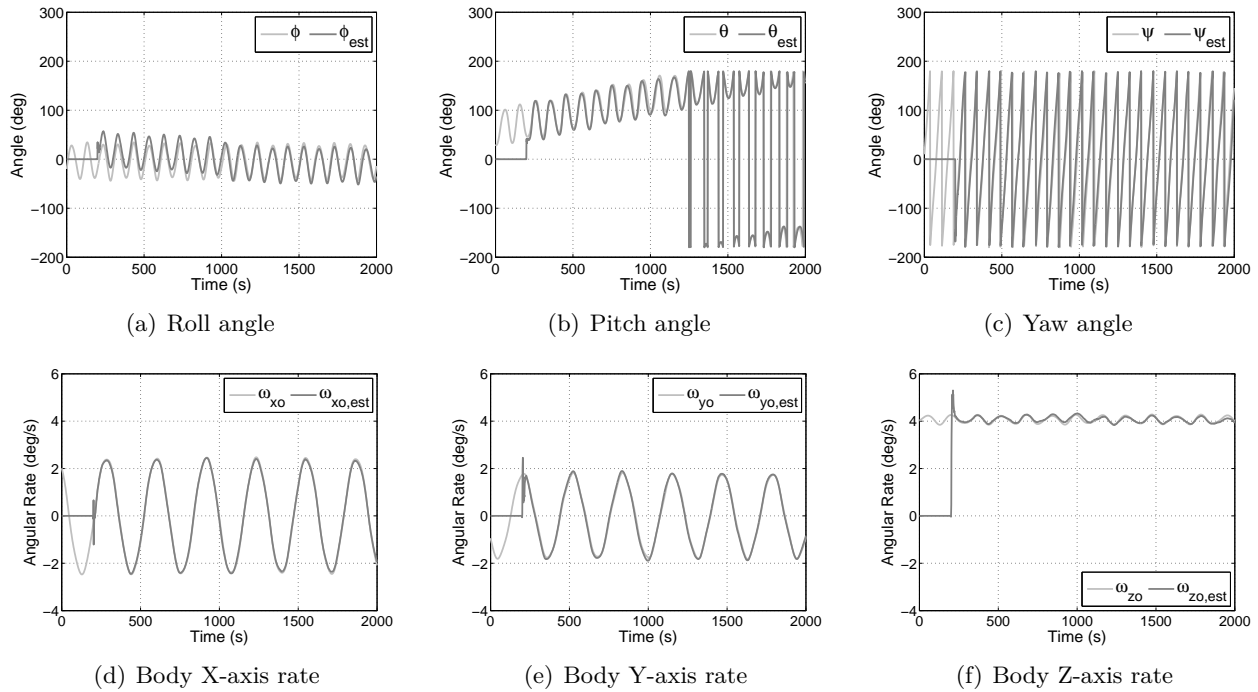
Figure 5.5 – The EKF performance in an uncontrolled tumbling state
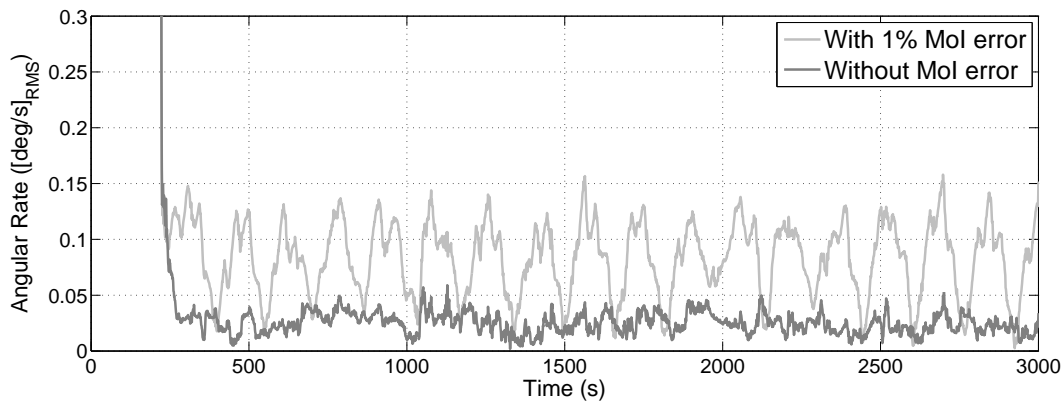
Figure 5.6 – The effect of MoI uncertainty on the EKF during an uncontrolled tumble

Controlled Stable State

Figure 5.7 shows the true and estimated Euler angles and estimated angular rates of the EKF during a controlled stable state. After approximately 50 seconds the estimation errors remain within ± 90 arcsec and ± 20 arcsec/s respectively. The star-tracker-based EKF can thus provide a position estimate that is within the required margin of 118 arcsec from Chapter 3. The 1-σ errors for the last 2500 seconds of the simulation are shown in Table 5.1. The star tracker is located in the satellite body YZ-plane, which implies that the around-boresight noise (which is larger than the across-boresight noise) will only affect the Y-axis and Z-axis estimates, hence the smaller X-axis 1-σ errors in Table 5.1.

	X-axis	**Y-axis**	**Z-axis**
σ_{angle} (arcsec)	4.9	23.3	24.7
σ_{rate} (arcsec/s)	2.4	3.8	4.5

Table 5.1 – The 1-σ estimation errors of the EKF in a controlled stable state

(a) Roll angle   (b) Pitch angle   (c) Yaw angle

(d) Body X-axis rate   (e) Body Y-axis rate   (f) Body Z-axis rate
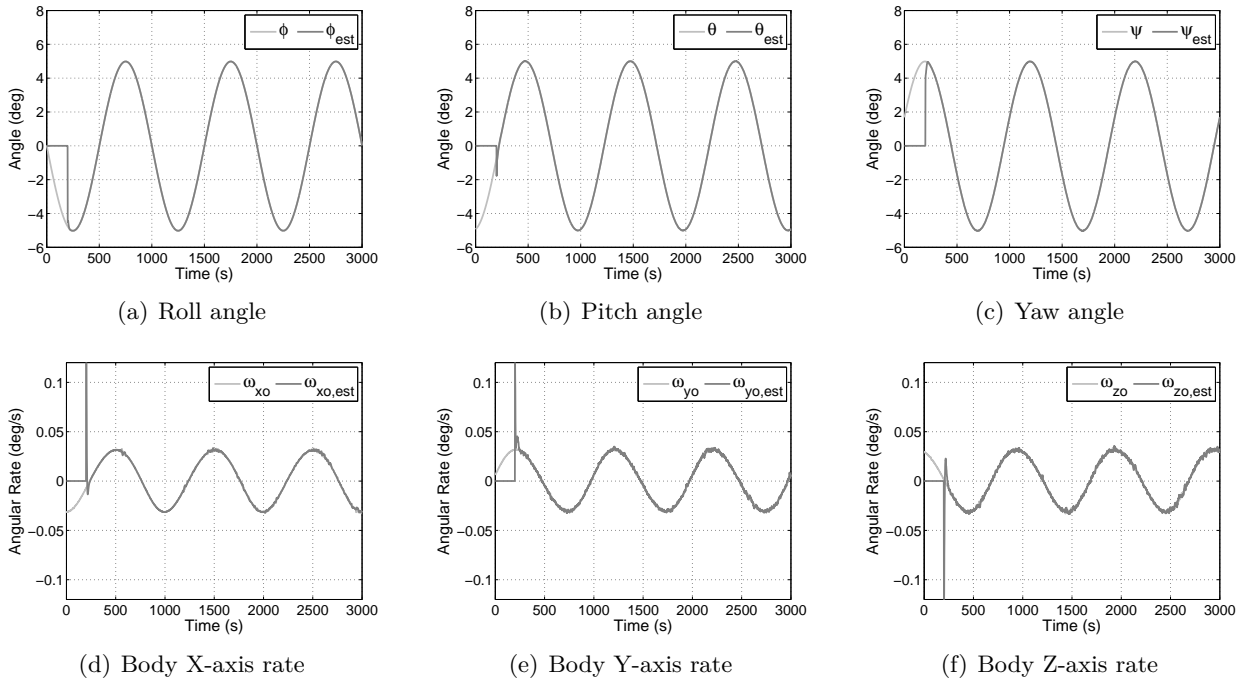
Figure 5.7 – The EKF performance in a controlled stable state

5.5   Summary

This chapter investigated several attitude determination methods by focusing on derivations, implementations and simulation results. The estimators that were considered for this project was the TRIAD and Optimized TRIAD algorithms, an RKF and a full state EKF.

Although TRIAD proved to be easy to implement, Optimized TRIAD provided slightly better attitude estimates with a small increase in computational load. A fundamental flaw of TRIAD and Optimized TRIAD is however that the quaternion estimates are directly influenced by sensor noise.

The magnetometer-based RKF was found to be a robust rate estimator which can be extremely useful during de-tumbling, where highly accurate estimates are not required. The RKF was furthermore simulated during a stable satellite state and proved to be relatively accurate given the simple system model.

The EKF was lastly investigated by simulating both an uncontrolled tumbling state and a controlled stable state. Although the magnetometer-based EKF provided more accurate rate estimates than the RKF did during an uncontrolled tumble, the EKF is more likely to be used during high-accuracy

manoeuvres. Even in the presence of MoI uncertainty, the star-tracker-based EKF produced estimates with arcsecond accuracies that were well within the given requirements.

It can thus be concluded that although Optimized TRIAD and the RKF are able to provide moderately accurate estimates (and are thus useful if such estimates are adequate), the EKF is a much more superior estimator and is suitable for high-accuracy applications.

Chapter 6

Attitude Control

6.1 Overview

There are few satellites in operation that do not require some form of attitude control, be it passive or active. The control of a satellite's attitude is often directly related to the main mission goals. For example, an Earth observation satellite requires precision pointing to ensure accuracy and stability during imaging manoeuvres. As a result of this requirement for EO satellites, only active control methods were investigated for this project.

Based on the hardware selection in Chapter 3, both magnetic and wheel control techniques were considered for active attitude controllers. The magnetic controllers are useful when precision pointing is not required (e.g. when in a safe mode or de-tumbling), and the wheel controllers are employed when higher accuracy is needed (e.g. during imaging).

This chapter will elaborate on the implementation and the working of various magnetic and wheel controllers. The magnetic controllers that were considered for this project include the B-dot, Y-spin, Cross-Product, Precession, and Momentum Dumping controllers. The wheel controllers that were considered include a Y-momentum controller and a Quaternion Feedback controller.

The working of the each attitude controller was furthermore verified through simulation and the results thereof will be shown. The effect of actuator noise was also investigated.

6.2 Magnetic Control

Active magnetic control on-board satellites are usually performed using three orthogonal magnetic torquers. The torque $\mathbf{N}_m$ generated by the magnetic torquers is a non-linear function of the magnetic dipole moment $\mathbf{M}$ and the local geomagnetic field vector $\mathbf{B}_b$ (in SBC), i.e.

$$\mathbf{N}_m \;\; = \;\; \mathbf{M} \times \mathbf{B}_b \; . \tag{6.2.1}$$

The output of each magnetic controller is $\mathbf{M}$, which will react with $\mathbf{B}_b$ to produce $\mathbf{N}_m$. Equation 6.2.1 furthermore implies that N_{mx} can only be generated by M_y or M_z. Similarly, N_{my} can only be generated by M_x or M_z and N_{mz} can only be generated by M_x or M_y.

As mentioned in Section 4.5, the magnetic dipole moment is subject to certain constraints. $\mathbf{M}$ must periodically be zeroed so that the magnetometer can be sampled and $\mathbf{M}$ also has a maximum value that depends on the specific torquer being used. The simulations behind the results shown in this section included the above-mentioned constraints.

The simulated controllers were furthermore fed by the estimated attitude and rates as opposed to the true attitude and rates and the magnetic control sample period was chosen as 1 s. The torquer on-time resolution was set to 10 ms and the maximum on-time was 800 ms. Lastly, the maximum dipole moment for all three magnetic torquers was set to 1 Am^2.

6.2.1   B-dot Controller

The B-dot controller [22] is a low-computational-load magnetic controller that does not require any knowledge of the satellite's current position, attitude or angular rates. The main goal of this controller is to align the satellite body Y-axis with the orbit normal vector. This process will also damp the X-axis and Z-axis angular rates, ω_{xo} and ω_{zo}.

The B-dot controller is executed by pulsing only the Y-axis torquer, where M_y is chosen to be directly proportional to the time derivative of β, the angle between B_{yb} and $\mathbf{B}_b$. If the derivative gain is K_d, then the B-dot controller can be expressed as

$$M_y \;=\; K_d\dot{\beta}\;, \tag{6.2.2}$$
$$\text{where}\quad \beta \;=\; \cos^{-1}\left(\frac{B_{yb}}{\|\mathbf{B}_b\|}\right)\;.$$

A modified version of the B-dot controller, which also uses the angle between B_{yo} and $\mathbf{B}_o$ in the process of calculating M_y, was presented by Stephen Hodgart in 1994 [23]. The modified controller is given by

$$M_y \;=\; K_d|\dot{\beta}-\dot{\alpha}|\;\text{sgn}(\dot{\beta}-\dot{\alpha})\;, \tag{6.2.3}$$
$$\text{where}\quad \alpha \;=\; \cos^{-1}\left(\frac{B_{yo}}{\|\mathbf{B}_o\|}\right)\;.$$

It is claimed that the modified B-dot controller shows improved performance compared to the original B-dot controller. Section 6.2.1.2 will show results of simulations that aimed to compare the two controllers' performance from an uncontrolled tumble.

6.2.1.1 B-dot Implementation

The main advantage of the B-dot controller is its simplicity. It is extremely easy to implement and does not require data from an estimator or the IGRF model to execute. A disadvantage of the modified B-dot controller is thus that it requires knowledge of the modelled orbit-referenced magnetic field to execute. Nonetheless, both controllers can be implemented with only a few steps.

The first step of the controllers is to calculate the discrete time derivative(s) of β (and α). If T_s is the sample period of the magnetic control loop, then

$$\dot{\beta}_k = \frac{\beta_k - \beta_{k-1}}{T_s} \tag{6.2.4}$$

$$\dot{\alpha}_k = \frac{\alpha_k - \alpha_{k-1}}{T_s} \ . \tag{6.2.5}$$

It should be noted from Equations 6.2.4 and 6.2.5 that β_{k-1} and α_{k-1} must be initialised to zero before executing the controllers for the first time.

Once the respective time derivatives have been calculated, the controllers can be executed by using Equations 6.2.2 and 6.2.3 and setting $M_x = M_z = 0$. The derivative gain K_d must be determined empirically with a focus on torquer on-time.

6.2.1.2 B-dot Simulation Results

The B-dot and modified B-dot controllers were simulated under identical circumstances, firstly to verify their working and secondly to compare their performance. The performance of a de-tumbling magnetic controller should be measured by calculating the total on-time of the torquers. The total on-time for each simulation was calculated from the start of the simulation until the angular rates had stabilised.

The initial angular rates (in IRC) and Euler angles were once again set to $\boldsymbol{\omega}_B^I = \begin{bmatrix} 2 & -1 & 4 \end{bmatrix}^T$ °/s and $\mathbf{RPY} = \begin{bmatrix} -20 & 30 & 10 \end{bmatrix}^T$ °. The respective controller (B-dot or modified B-dot) was switched on after 1000 s.

Figures 6.1 and 6.2 show the orbit-referenced angular rates of the satellite for the B-dot controller and the modified B-dot controller respectively. In both cases the angular rates stabilise after approximately 6000 s. It is evident from these figures (and also from various other simulations of which the results are not shown) that the modified B-dot controller offers no noticeable improvement in terms of settling time for the given orbit, satellite, and initial conditions.

The total on-time of the magnetic torquers (from controller activation until the rates settled) for the simulations depicted in Figures 6.1 and 6.2 were calculated as 3987 s for the B-dot controller and 3990 s for the modified B-dot controller.
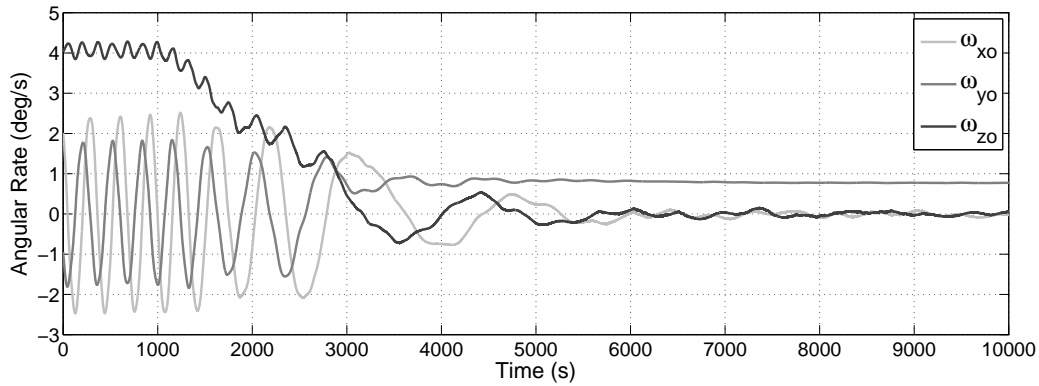
Figure 6.1 – The working of the B-dot magnetic controller
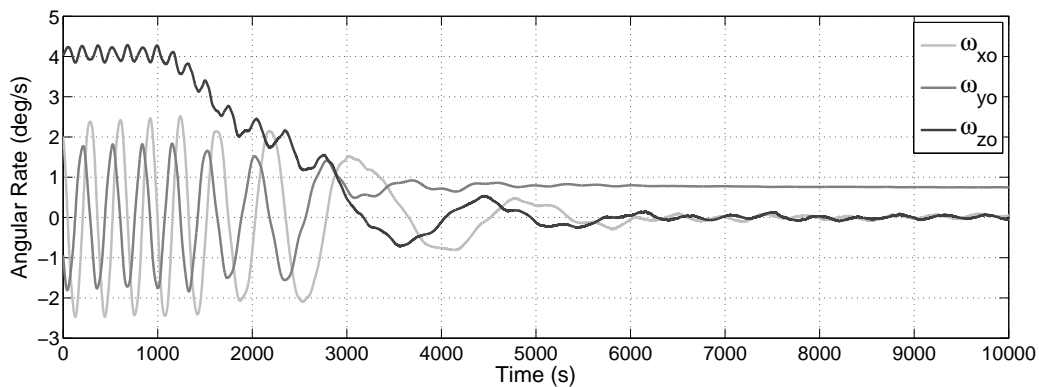
Figure 6.2 – The working of the modified B-dot magnetic controller

Simulations furthermore showed that the B-dot controller offers superior nutation damping to the modified B-dot controller. For the purpose of de-tumbling, the original B-dot controller seems to be more suitable than the modified B-dot controller. The lack of improvement offered by the modified controller and the added computational complexity thereof strengthens the above-mentioned argument.

Another interesting observation made from simulating the B-dot controller was that the value at which ω_{yo} settles is arbitrary. It was found that the settling value changes as the activation time of the controller changes. Figure 6.3 shows ω_{yo} for three different activation times of the B-dot controller and it is clear from this figure that the angular rate at which ω_{yo} settles is arbitrary. A logical explanation for this phenomenon is that only the Y-axis magnetic torquer is used for the B-dot controller, which means that only the X-axis and Z-axis rates are directly affected by the controller.

6.2.2   Y-spin Controller

The Y-spin magnetic controller has the function of bringing the Y-spin rate of the satellite ω_{yo} to a desired value $\omega_{yo,ref}$ and it is often combined with the B-dot controller [24; 15; 25]. The controller is implemented by using either the X-axis or the Z-axis magnetic torquer, depending on the magnitudes
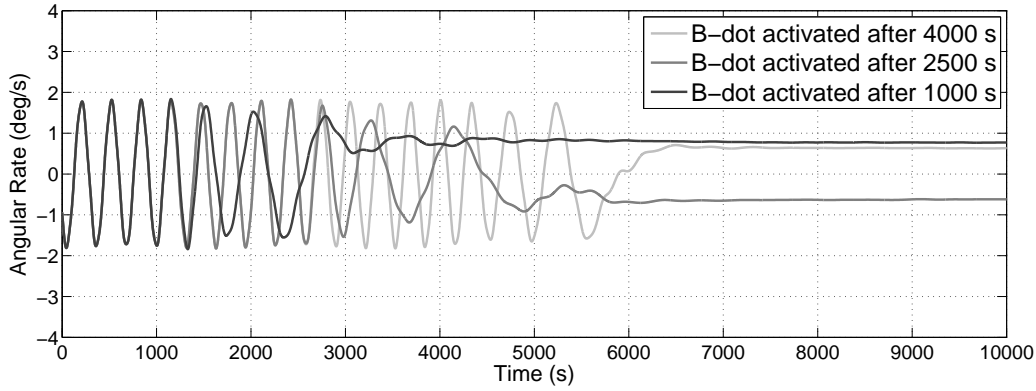
Figure 6.3 – An investigation into the Y-rate settling value for the B-dot controller

of B_{xb} and B_{zb}, and can be expressed as

$$M_x \quad = \quad K_s(\hat{\omega}_{yo} - \omega_{yo,ref})\,\mathrm{sgn}(B_{bz}) \qquad \text{if } |B_{bz}| > |B_{bx}| \tag{6.2.6}$$

$$M_z \quad = \quad K_s(\omega_{yo,ref} - \hat{\omega}_{yo})\,\mathrm{sgn}(B_{bx}) \qquad \text{if } |B_{bx}| \geq |B_{bz}| \ , \tag{6.2.7}$$

where K_s is the controller gain. It is evident from Equations 6.2.6 and 6.2.7 that the Y-spin controller requires knowledge of $\hat{\omega}_{yo}$ to execute. The Rate Kalman Filter (or a similar estimator or a rate sensor) should therefore be implemented alongside the Y-spin controller.

6.2.2.1   Y-spin Implementation

The Y-spin controller is implemented as part of the de-tumbling process. However, before the Y-spin controller can be activated, the rate estimator must be allowed to converge. As a result of the fact that the Y-spin controller is mostly used together with the B-dot controller, the rate estimator will be the first activated process when de-tumbling a satellite. Once the rate estimator has converged, the two magnetic controllers can be activated. It should however be noted that when M_x is generated, M_z should be zeroed. Similarly, when M_z is generated, M_x should be zeroed. The value of K_s must lastly be determined empirically with the main goal of minimising the torquer on-time.

It should furthermore be noted that a stable Y-spin is only possible if I_{yy} is the largest moment of inertia (and the products of inertia are small compared to the moments of inertia).

6.2.2.2   Y-spin Simulation Results

The Y-spin magnetic controller was simulated under conditions where the B-dot controller and a magnetometer-based RKF had already been active for a long time. In fact, the simulation shown in this section is a continuation of the B-dot simulation shown in Figure 6.1. The Y-spin controller was activated 1000 s after the start of the simulation with a desired rate of $\omega_{yo,ref} = -2°/\mathrm{s}$.

Figure 6.4(a) illustrates the alignment of the body Y-axis (Y_B) with the orbit Y-axis (Y_O), as is the aim of the B-dot and Y-spin controllers. The X-axis and Z-axis were however still subject to nutation.

The orbit-referenced angular rate of the satellite during the simulation is shown in Figure 6.4(b), where it can be seen that ω_{yo} reached the desired rate of $-2°/s$ approximately 4500 s after the Y-spin controller was activated. The settling time of the controller is dependent on the gain K_s.
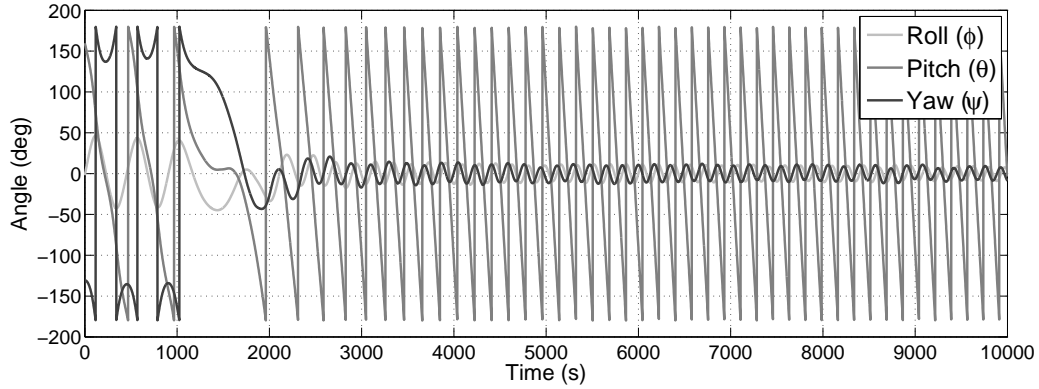
(a) Euler angles
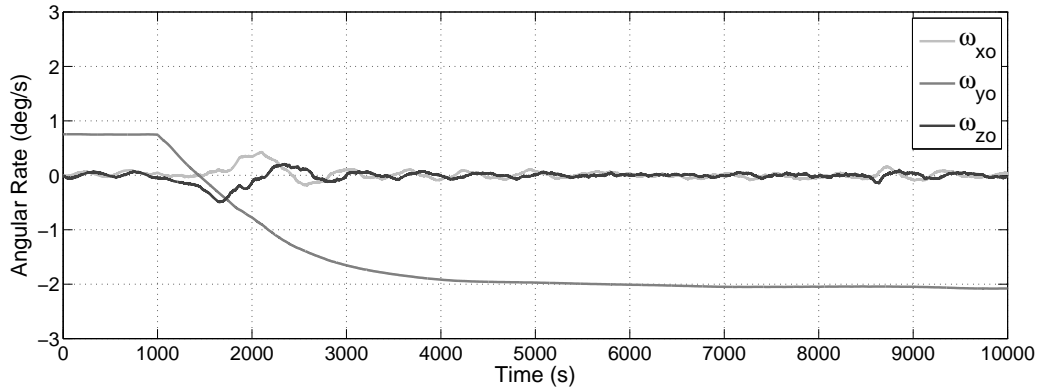
(b) Orbit-referenced angular rates

Figure 6.4 – The working of the Y-spin magnetic controller

6.2.3   Cross-Product Controller

The Cross-Product magnetic controller is based on the vector cross-product operation. The controller involves defining a control error vector $\mathbf{e}$, which must be controlled to zero by a certain required torque $\mathbf{N}_{m,req}$, which is related to $\mathbf{e}$ via

$$\mathbf{N}_{m,req} = -\mathbf{e} . \tag{6.2.8}$$

The magnetic moment $\mathbf{M}$ required to generate $\mathbf{N}_{m,req}$ can be determined by firstly substituting Equation 6.2.8 into Equation 6.2.1, which delivers

$$-\mathbf{e} = \mathbf{M} \times \mathbf{B}_b , \tag{6.2.9}$$

and by secondly performing mathematical manipulation on Equation 6.2.9 (see Appendix C), which produces

$$\mathbf{M} \quad = \quad \frac{\mathbf{e} \times \mathbf{B}_b}{\|\mathbf{B}_b\|^2} \; . \tag{6.2.10}$$

The Cross-Product controller can also be modified to be less dependent on the magnitude of the geomagnetic field vector. The control law for calculating $\mathbf{M}$ then becomes

$$\mathbf{M} \quad = \quad \frac{\mathbf{e} \times \mathbf{B}_b}{\|\mathbf{B}_b\|} \; . \tag{6.2.11}$$

The Cross-Product controller will however always be a non-ideal controller as a result of the cross-coupling between axes when performing magnetic control.

6.2.3.1 Cross-Product Implementation

The Cross-Product controller can be implemented in numerous different ways, of which only two will be shown. All of these scenarios have one thing in common: the control error vector $\mathbf{e}$ must be determined before the magnetic dipole moment $\mathbf{M}$ can be calculated.

The Cross-Product controller can, for example, be implemented as a de-tumbling controller (instead of the B-dot and Y-spin controllers), where $\mathbf{e}$ is defined as

$$\mathbf{e} \quad = \quad \begin{bmatrix} K_{dx}\hat{\omega}_{xo} \\ K_{dy}(\hat{\omega}_{yo} - \omega_{yo,ref}) \\ K_{dz}\hat{\omega}_{zo} \end{bmatrix} \; . \tag{6.2.12}$$

The downside of using this controller for de-tumbling is that although the reference Y-rate is followed, the alignment of the body Y-axis with the orbit Y-axis takes significantly longer than with the B-dot and Y-spin controllers. Once again, a stable Y-axis spin is only possible if I_{yy} is the largest moment of inertia (and the products of inertia are small compared to the moments of inertia).

The Cross-Product controller can also be used in conjunction with a Y-momentum wheel controller to damp ω_{xo} and ω_{zo} and to keep the Y-wheel angular momentum h_{wy} at the desired reference value $h_{wy,ref}$. For the above-mentioned implementation, $\mathbf{e}$ is defined as

$$\mathbf{e} \quad = \quad \begin{bmatrix} K_{dx}\hat{\omega}_{xo} \\ K_{dy}(h_{wy,meas} - h_{wy,ref}) \\ K_{dz}\hat{\omega}_{zo} \end{bmatrix} \; . \tag{6.2.13}$$

It should be noted that both Equations 6.2.12 and 6.2.13 require rate estimates to execute. Lastly, the gains of the Cross-Product controller need to be determined empirically.

6.2.3.2   Cross-Product Simulation Results

The two implementations of the Cross-Product controller were tested through simulation. Before a controllers was switched on, the appropriate estimator was activated and allowed to converge. The controller being simulated was switched on 1000 s after the start simulation.

The de-tumbling Cross-Product controller was simulated with initial angular rates of $\boldsymbol{\omega}_B^I = \begin{bmatrix} 2 & -1 & 4 \end{bmatrix}^T °/s$ and Euler angles of $\mathbf{RPY} = \begin{bmatrix} -20 & 30 & 10 \end{bmatrix}^T °$. Figure 6.5 shows the orbit-referenced angular rates for a de-tumbling process with a reference Y-rate of $\omega_{yo,ref} = -2°/s$. The reference rate is reached after approximately 4000 s. The oscillations in ω_{xo} and ω_{zo} have a period of about 177 s, which correlates with a frequency of approximately $2°/s$. The oscillations can therefore be attributed to gyroscopic coupling.
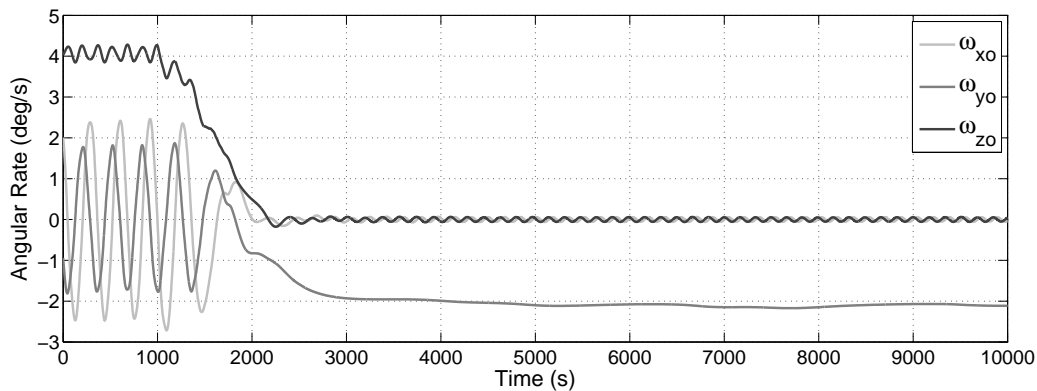
Figure 6.5 – The working of the de-tumbling Cross-Product magnetic controller

The Y-momentum Cross-Product controller needed to be simulated with the Y-momentum wheel controller. The satellite was in a stable Y-spin at the start of the simulation, after which both the Y-momentum magnetic controller and the Y-momentum wheel controller were activated at 1000 s. The angular momentum that was stored in the Y-axis wheel, h_{wy}, can be seen in Figure 6.6, which clearly indicates that the controller keeps h_{wy} at its reference value of $h_{wy,ref} = -15$ mNms.
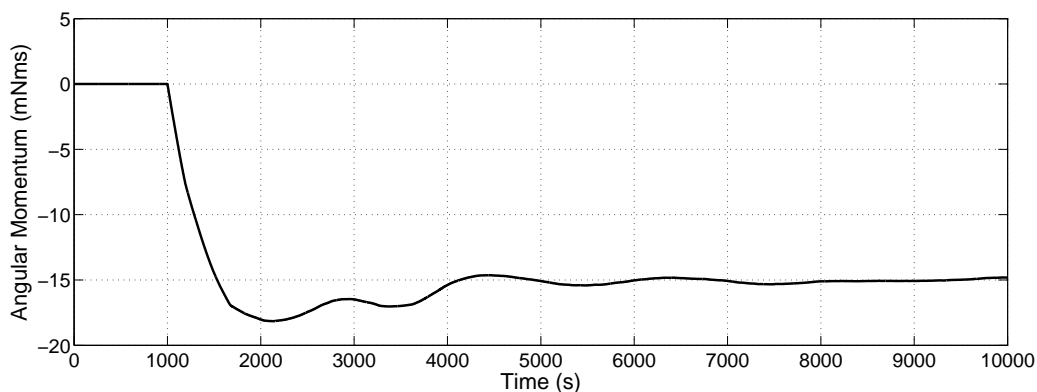
Figure 6.6 – The working of the Y-momentum Cross-Product magnetic controller

Figures 6.5 and 6.6 clearly prove the ability of the Cross-Product magnetic controller to serve as either a de-tumbling controller or a Y-momentum management controller.

6.2.4   Precession Controller

The Precession magnetic controller is used to precess (or rotate) an angular momentum vector. This controller is especially useful when the satellite is in a Y-Thomson spin [26]. For example, if the satellite's main solar panel is on the $-$Y-facet, the Precession controller can be used to point the solar panel roughly towards the sun to generate near-maximum power (which is extremely helpful when the satellite is in a power saving mode).

The sun-seeking Precession controller [15] is based on the Cross-Product magnetic controller (see Section 6.2.3). The control error vector $\mathbf{e}$ for the Precession controller is based on the precession vector $\mathbf{\Omega}_p$, which is defined as

$$\mathbf{\Omega}_p \;=\; K(\mathbf{u}_{sp}^{SBC} \times \mathbf{S}_b) \;, \tag{6.2.14}$$

where K is the precession gain, $\mathbf{u}_{sp}^{SBC}$ is the unit vector indicating the main solar panel's position and $\mathbf{S}_b$ is the measured unit sun vector. If the Y-axis body angular momentum vector (as a result of the Y-spin) is $\mathbf{h}_{by}$, then $\mathbf{e}$ can be defined as

$$\mathbf{e} \;=\; \mathbf{h}_{by} \times \mathbf{\Omega}_p \;. \tag{6.2.15}$$

The required magnetic dipole moment can then be calculated using Equation 6.2.11.

6.2.4.1   Precession Implementation

The Precession controller requires knowledge of $\mathbf{h}_{by}$ and $\mathbf{S}_b$ to execute. Given that

$$\mathbf{h}_{by} \;=\; \begin{bmatrix} 0 & I_{yy}\omega_{yo,ref} & 0 \end{bmatrix}^T \;, \tag{6.2.16}$$

a rate estimator (or rate sensor) is required to calculate $\mathbf{h}_{by}$. A sun sensor (most likely a CSS) is also required to provide the controller with $\mathbf{S}_b$.

The nature of this controller furthermore implies that it can only be activated in the sunlit part of the orbit. The controller that brought about the Y-spin should thus be executed during eclipse to keep the Y-axis spin rate at the desired value.

The value of K must be directly proportional to β'_{sp}, the angle between $\mathbf{u}_{sp}^{SBC}$ and $\mathbf{S}_b$, thus

$$K \;=\; k_p \beta'_{sp} \;. \tag{6.2.17}$$

The gain k should be determined empirically with the goal of letting β'_{sp} drop below $20°$ within approximately three orbits [15].

6.2.4.2 Precession Simulation Results

The working of the sun-seeking Precession controller was verified by simulating a scenario where the satellite had already been brought to a Y-Thomson spin. A combination of the Precession controller (during the sunlit part of the orbit) and the B-dot and Y-spin controllers (during eclipse) was then activated after 1000 s.

Figure 6.7 shows S_{by}, the Y-axis component of the unit sun vector, as measured by the CSS for the simulation at hand. The initial value is approximately -0.58, which relates to a sun incidence angle (on the main solar panel) of $55°$ and power generation of 58%. Once the Precession controller is activated, the incidence angle drops to approximately $18°$ (with $S_{by} \approx -0.95$), which boosts the power generation to 95%.
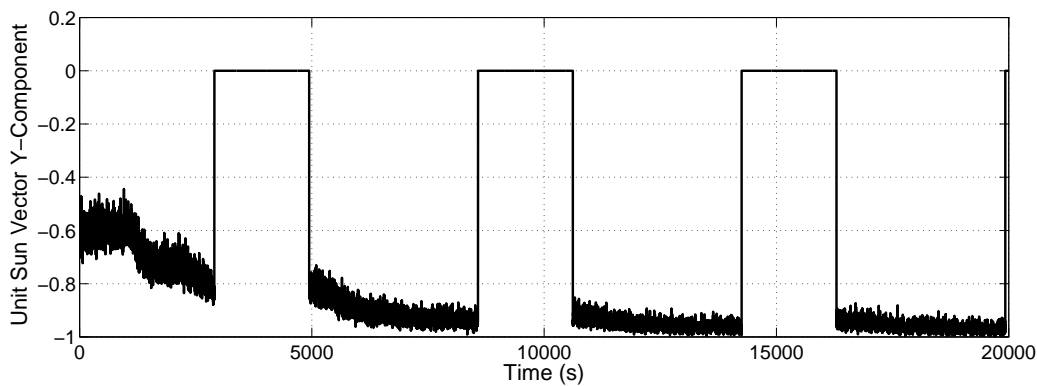
Figure 6.7 – The working of the Precession magnetic controller

6.2.5 Momentum Dumping Controller

A reaction wheel used on-board a satellite has a limited angular momentum storage capability. Furthermore, the higher the angular rate of a wheel, the more power it consumes and the more disturbance torque is generated by the static and dynamic imbalances in the wheel. All of the above-mentioned statements indicate that a need exists to manage the angular momentum stored in reaction wheels.

The use of magnetic torquers is a "cheap and effective means" [16] of dumping angular momentum. Although there are various complex momentum management controllers in existence [16], the Momentum Dumping controller that was considered for this project is a simple magnetic controller based on the Cross-Product controller (see Section 6.2.3) [27].

The control error vector **e** is defined to be directly proportional to the difference between the current angular momentum $\mathbf{h}_w$ and the desired angular momentum $\mathbf{h}_{w,ref}$. If K_w is a positive gain, then

$$\mathbf{e} = -K_w(\mathbf{h}_w - \mathbf{h}_{w,ref}) \ . \tag{6.2.18}$$

For the Momentum Dumping controller, the required magnetic dipole moment **M** should not be calculated with Equation 6.2.11, but rather with Equation 6.2.10, or

$$\mathbf{M} \quad = \quad \frac{\mathbf{e} \times \mathbf{B}_b}{\|\mathbf{B}_b\|^2} \; .$$

6.2.5.1   Momentum Dumping Implementation

As a first rule of implementation, the Momentum Dumping magnetic controller should not be used when the satellite is performing high-accuracy manoeuvres. An EO satellite with a visual spectrum payload, for example, should only employ the Momentum Dumping controller during eclipse, when the satellite is in a stable nadir-pointing attitude.

The control error **e** will naturally be calculated using the measured angular momentum $\mathbf{h}_{w,meas}$, thus Equation 6.2.18 becomes

$$\mathbf{e} \quad = \quad -K_w(\mathbf{h}_{w,meas} - \mathbf{h}_{w,ref}) \; . \tag{6.2.19}$$

The gain K_w determines the bandwidth of the controller (i.e. how fast it dumps the angular momentum). K_w should thus be chosen to meet the required dumping specifications, for example, the controller must be able to dump 5 mNms per orbit.

6.2.5.2   Momentum Dumping Simulation Results

The Momentum Dumping controller was simulated together with the Eigenaxis wheel controller, which kept the satellite in a stable nadir-pointing attitude. The wheel controller had been on for a long time before the start of the simulation results shown in this section. The Momentum Dumping controller was then switched on after 1000 s with $\mathbf{h}_{w,ref} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ Nms.

Figure 6.8 shows the resulting true angular momentum. h_{wx}, h_{wy}, and h_{wz} can clearly be seen to decrease as a result of the Momentum Dumping controller. Figure 6.8 furthermore illustrates that the controller was able to dump approximately 14 mNms in 9000 s in the Y-axis.
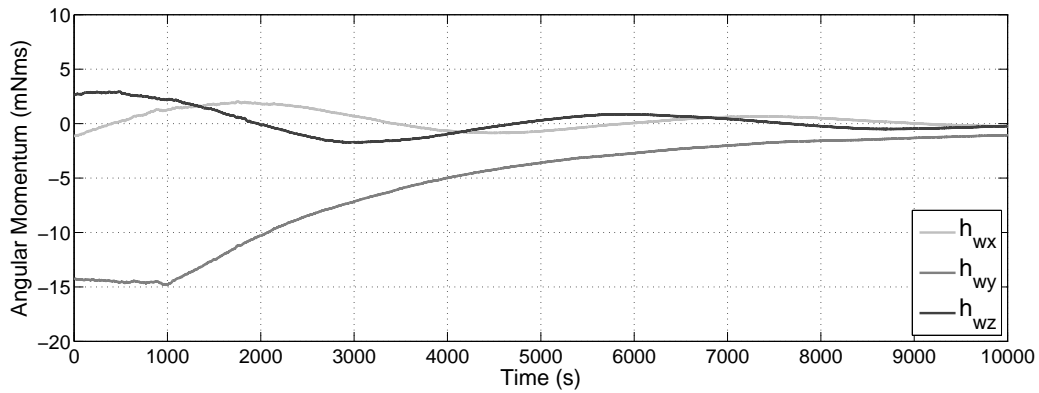
Figure 6.8 – The working of the Momentum Dumping magnetic controller

6.3   Wheel Control

Wheel control on-board satellites can be performed using either momentum wheels or reaction wheels. Momentum wheels are usually implemented to provide gyroscopic stiffness, whereas reaction wheels are often employed for precision pointing, for example on an imaging satellite.

A wheel controller outputs a required torque $\mathbf{N}_{w,req}$, which is then integrated to determine $\mathbf{h}_{w,req}$, the required angular momentum. $\mathbf{h}_{w,req}$ is converted to a reference angular rate which the internal wheel speed controller will follow (see Section 4.5.2). The wheel controller in turn requires knowledge of the estimated satellite attitude and rates to execute. The physical limitations of the wheel (i.e. maximum torque and angular momentum, $\mathbf{N}_{w,max}$ and $\mathbf{h}_{w,max}$) must also be included in the wheel controller. Figure 6.9 illustrates the wheel control loop inside the ADCS.

Figure 6.9 – An illustration of the ADCS wheel control loop

The results that will be shown in this section are based on the wheel dynamics discussed in Section 4.5. $\mathbf{N}_{w,max}$ and $\mathbf{h}_{w,max}$ were respectively set to 5 mNm and 60 mNms. Although the wheel control sample period for the simulations was chosen as $T_s = 1$ s, the commanded change in angular momentum (which is calculated from the commanded torque) was broken up into 10 smaller steps (i.e. $\frac{T_s}{T_w}$) to miti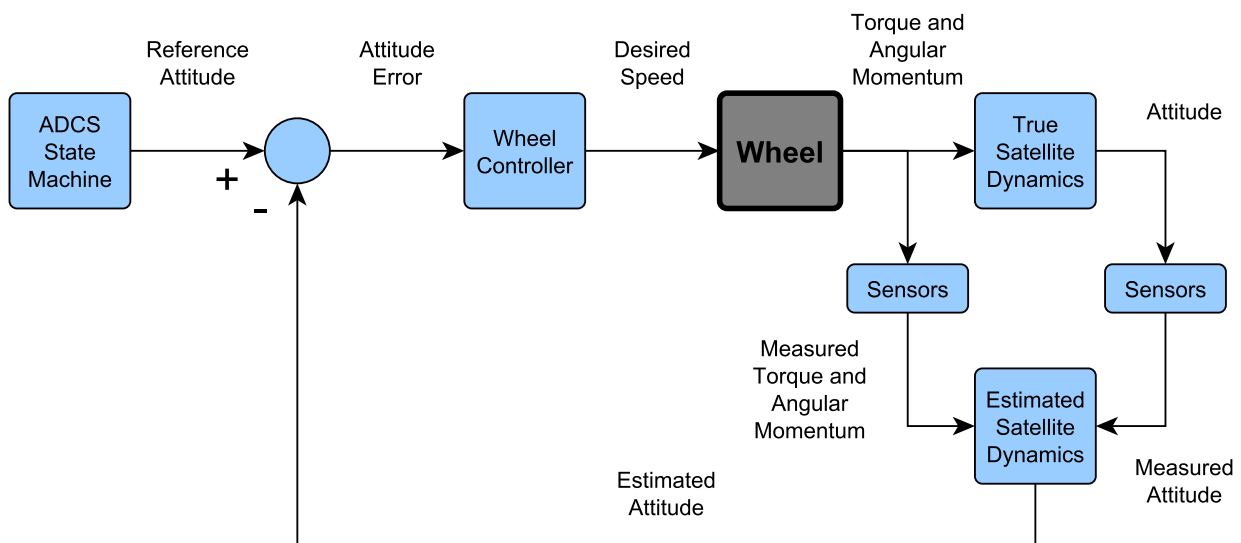gate the effect of the wheel speed overshoot (see Chapter 4). The speed command would then be in the form of a ramp rather than a step. This technique can be implemented practically on a satellite, which means that the credibility of the simulation environment's results will not be affected.

6.3.1 Y-momentum Controller

The satellite will enter a stable Y-spin during de-tumbling, as mentioned in Section 6.2.2. The Y-momentum wheel controller [24] has the purpose of absorbing the Y-axis body angular momentum (as a result of the Y-spin) and controlling the pitch angle to a certain reference value θ_{ref}. The required Y-wheel torque $N_{wy,req}$ is calculated by a PD-type control law, which is defined as

$$N_{wy,req} \quad = \quad K_p(\hat{\theta} - \theta_{ref}) + K_d\hat{\omega}_{yo} \ , \tag{6.3.1}$$

where K_p is the proportional gain and K_d is the derivative gain. Equation 6.3.1 implies that the Y-momentum controller requires estimates of both the current attitude (or at least the pitch angle) and the current angular rates. A magnetometer-based EKF (see Section 5.4 or a simple pitch filter [24] can be employed for this purpose.

6.3.1.1 Y-momentum Implementation

Before the Y-momentum wheel controller can be switched on, the Y-axis of the satellite body needs to be aligned with the Y-axis of the orbit. An estimate of the roll and yaw angles are thus required. The estimator used with the Y-momentum controller will most likely be the first estimator in the de-tumbling process to provide an attitude estimate. The estimator thus also needs to be allowed to converge before the controller is activated.

The first term of Equation 6.3.1 will fall away when the pitch angle does not need to be controlled. The required torque will thus be

$$N_{wy,req} \quad = \quad K_d\hat{\omega}_{yo} \ . \tag{6.3.2}$$

Furthermore, the settling time of the Y-momentum wheel controller does not need to be very low, and the gains K_p and K_d should be chosen accordingly. To prevent angular momentum saturation, a Y-momentum management magnetic controller (see Section 6.2.3) should be employed to manage h_{wy}.

6.3.1.2   Y-momentum Simulation Results

The Y-momentum wheel controller was simulated as part of a de-tumbling scenario, where the satellite had already been brought to a Y-Thomson state with a reference rate of $\omega_{yo,ref} = -1°/s$. The controller was activated after 1000 s with $\theta_{ref} = 0°$ and a 1% settling time of 1000 s. At 10000 s the reference pitch angle was set to 90°.

Figure 6.10 shows the Euler angles and the orbit-referenced angular rates for the simulation at hand. The angular rate can be seen to settle near zero after 1000 s, after which the pitch angle remains within $\pm 2°$. After the reference pitch angle was set to $\theta_{ref} = 90°$, the pitch angle remained between 88° and 92°.
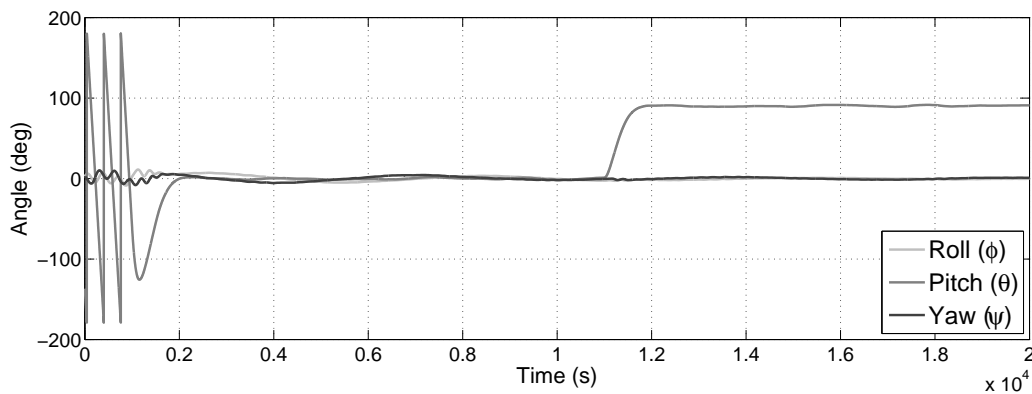
(a) Euler angles
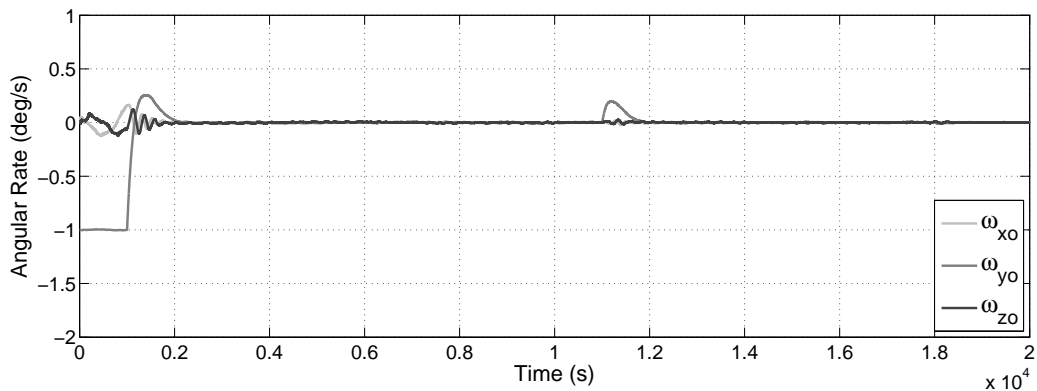
(b) Orbit-referenced angular rates

Figure 6.10 – The working of the Y-momentum wheel controller

6.3.2   Quaternion Feedback Controller

The globally stable Quaternion Feedback wheel controller [28] provides the satellite with the ability to control all three its axes accurately. The derivation of the controller is based on the Euler dynamic equation from Equation 1.3.24. If disturbances are ignored and it is assumed that the wheels are the

only actuators used, then the Euler dynamic equation becomes

$$\mathbf{J}\dot{\boldsymbol{\omega}}_B^I \;=\; -\mathbf{N}_w - \mathbf{N}_{gyro} \; . \tag{6.3.3}$$

The required torque $\mathbf{N}_{w,req}$ for the Quaternion Feedback controller is defined [15] as

$$\mathbf{N}_{w,req} \;=\; K_p \mathbf{J}\hat{\mathbf{q}}_e + K_d \mathbf{J}\hat{\boldsymbol{\omega}}_e - \hat{\boldsymbol{\omega}}_B^I \times \left(\mathbf{J}\hat{\boldsymbol{\omega}}_B^I + \mathbf{h}_{w,meas} \right) \; , \tag{6.3.4}$$

where K_p is the proportional gain, K_d is the derivative gain, $\hat{\mathbf{q}}_e$ is the quaternion error (see Section 1.3) between the estimated quaternion $\hat{\mathbf{q}}$ and the commanded quaternion $\mathbf{q}_c$, and $\hat{\boldsymbol{\omega}}_e$ is the estimated angular rate error. Substituting Equation 6.3.4 into Equation 6.3.3 delivers

$$\mathbf{J}\left(\dot{\boldsymbol{\omega}}_B^I + K_d \hat{\boldsymbol{\omega}}_e + K_p \hat{\mathbf{q}}_e \right) \;=\; 0 \; , \tag{6.3.5}$$

which almost resembles a linear second order system. The gains K_d and K_p can be related to the damping ζ and natural frequency ω_n of a desired second order system if an assumption is made. Firstly, let $\boldsymbol{\lambda}$ be a unit vector along the Eigenaxis. $\hat{\mathbf{q}}_e$ can now be expressed as

$$\hat{\mathbf{q}}_e \;=\; \sin\left(\frac{\Phi}{2} \right)\boldsymbol{\lambda} \; , \tag{6.3.6}$$

where Φ is the angle associated with the Eigenaxis rotation from $\hat{\mathbf{q}}$ to $\mathbf{q}_c$. If it is assumed [28] that the angular rate is small (thus the small angle approximation from Section 1.3 is valid), then Equation 6.3.5 becomes

$$\ddot{\Phi} + K_d \dot{\Phi} + K_p \frac{\Phi}{2} \;=\; 0 \;=\; \ddot{\Phi} + 2\zeta\omega_n \dot{\Phi} + \omega_n^2 \Phi \; . \tag{6.3.7}$$

Equation 6.3.7 clearly shows that K_d and K_p are related to ζ and ω_n through

$$K_d \;=\; 2\zeta\omega_n \tag{6.3.8}$$

$$\text{and} \quad K_p \;=\; 2\omega_n^2 \; . \tag{6.3.9}$$

Lastly, Equation 6.3.4 implies that attitude and rate estimates are required by the Quaternion Feedback controller. An Extended Kalman Filter (or a similar estimator) can be used for this purpose.

6.3.2.1   Quaternion Feedback Implementation

The sensors that are supplying the estimator with measurements will directly influence the accuracy of the Quaternion Feedback wheel controller, which means that different manoeuvres will require different sensors.

The damping and natural frequency (which is related to settling time) should primarily be chosen to satisfy any mission specifications (for example, a 30° slew must not take longer than 30 s). A second

objective in the choice of ζ and ω_n is to ensure that the wheels' torque or angular momentum are not saturated.

The values of ζ and ω_n may also vary for different scenarios. For example, if a satellite will only be in a nadir-pointing attitude during eclipse, the natural frequency can be lower than during the sunlit part of the orbit to lower the control effort.

6.3.2.2   Quaternion Feedback Simulation Results

The Quaternion Feedback wheel controller was simulated with optimal damping ($\zeta = 0.707$) and a 1% settling time of 30 s (to satisfy the agility specification from Section 3.2). At the start of the simulation the satellite was in an uncontrolled state with low angular rates. The Quaternion Feedback controller was activated after 300 s with a nadir-pointing command. The commanded Euler angles $\mathbf{RPY}_c$ were set to $\begin{bmatrix} 10 & 20 & 30 \end{bmatrix}^T$ ° at 600 s and back to zero at 900 s. The EKF discussed in Chapter 5 was employed as estimator and received measurements from the star tracker. No magnetic control was active during the simulation and the 1-σ wheel speed measurement noise was set to 0.5 rpm.

Figures 6.11(a) and 6.11(b) show the resulting Euler angles and orbit-referenced angular rates respectively. The 1-σ Euler angle and angular rate errors were calculated as $\begin{bmatrix} \sigma_\phi & \sigma_\theta & \sigma_\psi \end{bmatrix} = \begin{bmatrix} 19.6 & 34.1 & 44.7 \end{bmatrix}$ arcsec and $\begin{bmatrix} \sigma_{\omega x} & \sigma_{\omega y} & \sigma_{\omega z} \end{bmatrix} = \begin{bmatrix} 14.6 & 8.8 & 21.2 \end{bmatrix}$ arcsec/s respectively. The higher pitch and yaw angle errors were a result of the star tracker placement in the satellite body's YZ-plane.

The EKF model of the satellite dynamics uses the measured wheel speed to determine the wheel angular momentum $\mathbf{h}_w$ and torque $\mathbf{N}_w$. Although the process of differentiating a noisy signal (the measured wheel speed) to calculate the torque $\mathbf{N}_w$ might generate greater noise, the low bandwidth of the EKF will ensure that noise is not propagated into the estimated states. To verify this theory, the 1-σ wheel speed measurement noise was varied between 0 rpm and 2 rpm for an identical simulation as before. The resulting 1-σ Euler angle and angular rate errors can be seen in Figure 6.12.

Although the 1-σ errors in Figure 6.12 increase with the wheel speed measurement noise, the slope of the increase is low. It is thus clear that the EKF has the ability to filter out high frequency measurement noise, thereby allowing accurate pointing on a stable platform.
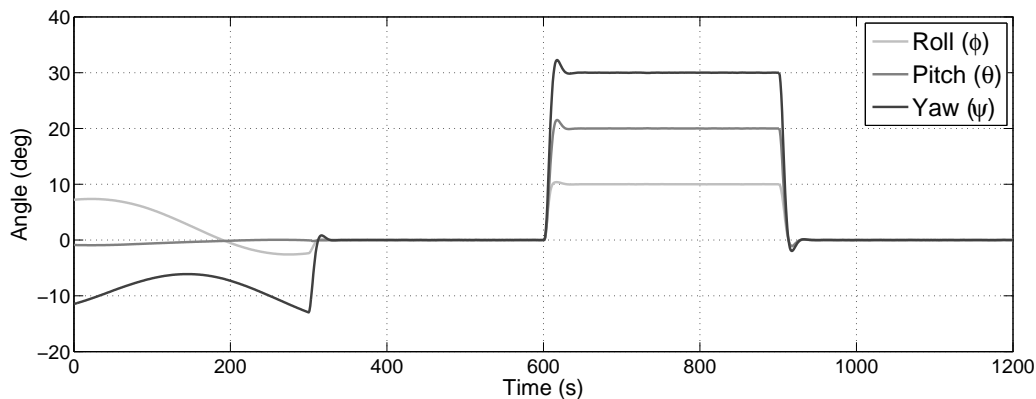
(a) Euler angles
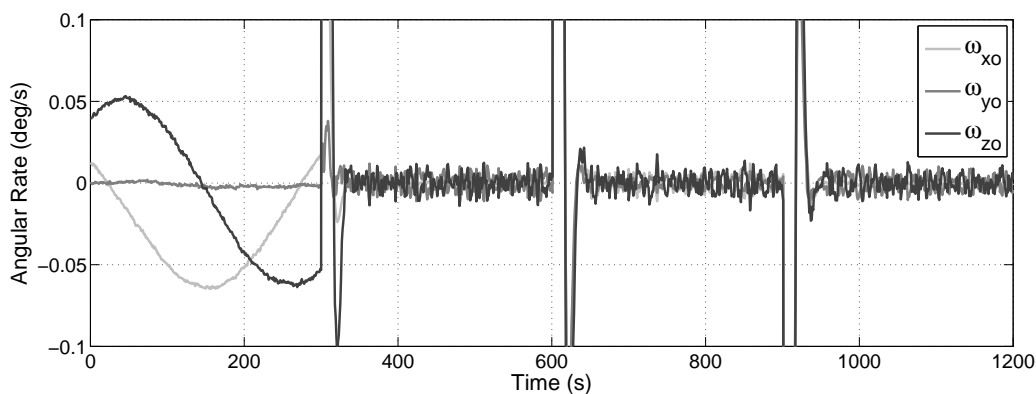
(b) Orbit-referenced angular rates

Figure 6.11 – The working of the Quaternion Feedback wheel controller

(a)

(b)

Figure 6.12 – The effect of wheel speed measurement noise on pointing accuracy and stability
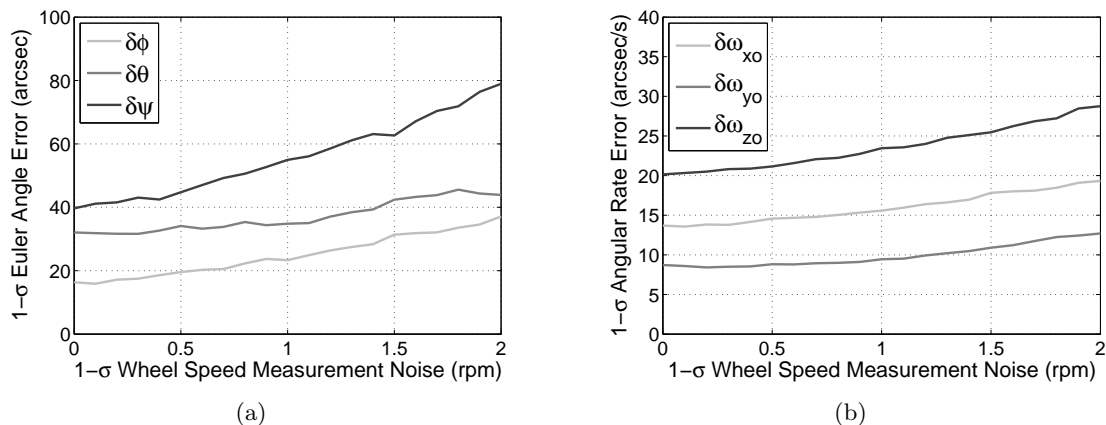
6.4   Summary

This chapter discussed only some of the many magnetic and wheel attitude controllers used on-board small LEO satellites. Basic derivations of the controller were given (where necessary), after which the practical implementation of each controller was discussed. The working of the controllers were also

verified through realistic simulations.

The modified B-dot magnetic controller was firstly not found to provide a noticeable improvement on the original B-dot controller. The combination of the B-dot and Y-spin was determined to be an acceptable de-tumbling controller when optimised for minimum magnetic torquer on-time. The Cross-Product controller proved to be quite versatile. It was firstly used as a de-tumbling controller and secondly as a Y-wheel momentum management tool. It was furthermore concluded that the Precession magnetic controller, which is based on the Cross-Product controller, can be implemented as a low-power sun-seeking controller. It was also found that the Momentum Dumping controller provided a simple but effective means to dump unwanted angular momentum stored in ADCS wheels.

Although the Y-momentum wheel controller could effectively bring the satellite to a stable nadir-pointing attitude (or even to a reference pitch angle), the Quaternion Feedback controller provided much higher pointing accuracy. It can lastly be concluded from the simulation results of the Quaternion Feedback controller that the aforementioned will be able to provide satisfactory stability and pointing accuracy for a variety of manoeuvres.

Chapter 7

An Application Specific ADCS

7.1   Overview

Any satellite mission requires an application specific ADCS solution to achieve its given goals. This unique ADCS solution is based on the main payload(s), mission constraints and performance specifications. Although an initial ADCS analysis (see Chapter 3) provides a good starting point for selecting ADCS hardware and possible estimation and control algorithms, an in-depth analysis is required to determine the performance of the application specific ADCS.

A set of modes can be defined for a satellite's ADCS to meet the given performance specifications in the presence of certain mission constraints (e.g. power). This chapter will firstly explain the choice of ADCS modes pertaining to this project and the state machine controlling the sequence in which these modes will be executed. The relevant estimators and controllers associated with each mode will also be discussed. For each of the ADCS modes, simulation results (based on the parameters mentioned in Chapters 2 and 4) and an analysis thereof will be given.

7.2   The ADCS State Machine

The ADCS developed for this project exists in the form of a state machine which aims to achieve the mission goals mentioned in Chapter 3. The states of the state machine are represented by the various ADCS modes, each having a unique combination of estimators, sensors, controllers and actuators.

Five ADCS modes were chosen for this project, namely Idle Mode, Safe Mode, Nominal Mode, FMC Imaging Mode, and Target Tracking Mode. No estimators or controllers will be active in Idle Mode and the satellite will only be in this mode when released from the launch vehicle. Safe Mode will not only be used to de-tumble the satellite initially, but also to serve as a fall-back mode in case of a system timeout. Nominal Mode entails tracking the sun during the sunlit part of the orbit and maintaining a nadir-pointing attitude during eclipse. FMC Imaging Mode will be executed as a result of a specific telecommand that schedules an imaging manoeuvre, whereas Target Tracking Mode will most likely be

activated as soon as a predefined ground-station (G/S) is in range. Although an imaging manoeuvre is the primary function of an EO satellite (and thus enjoys priority over Target Tracking Mode), target tracking plays an important role in the power budget and antenna choice of a satellite.

A basic flow diagram of the ADCS state machine can be seen in Figure 7.1.
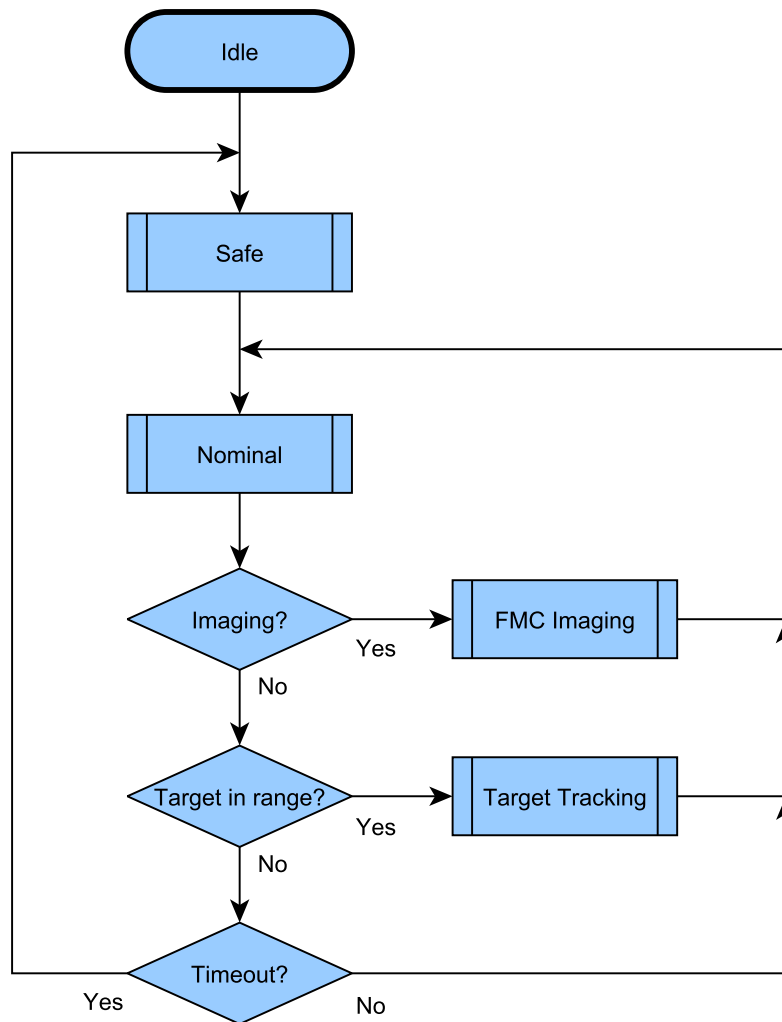
Figure 7.1 – A high-level model of the ADCS state machine

7.3   Idle Mode

When the satellite is released from the launch vehicle, the default ADCS mode will be Idle Mode. This is also the only time the satellite will be in this mode. None of the sensors or actuators will be used in this mode. Once it has been established (via ground station communication) that all the satellite components are functioning properly, the satellite will enter Safe Mode.

7.4 Safe Mode

The satellite will be in an uncontrolled tumble when released from the launch vehicle and it will only start to de-tumble once Safe Mode has been activated. Not all the facets of the satellite will have solar panels, which means that power consumption needs to be minimised in Safe Mode. The aim of this mode is thus to point the main solar panel of the satellite to the sun with the least amount of control effort. Safe Mode will also serve as a fall-back mode in certain erroneous satellite states, for example if no telecommands have been received for 3 days.

The first controller will need to bring the satellite's Y-rate to a certain reference rate while damping the X-rate and the Z-rate. Thereafter the main solar panel needs to be precessed towards the sun to maximise power generation. Once the solar panel is pointing roughly towards the sun (and sufficient power has been generated), the reaction wheels can be activated to absorb the body angular momentum and to ensure 3-axis stabilisation. The satellite will then be completely de-tumbled and Nominal Mode will be activated.

Either the Cross-Product or the B-dot and Y-spin magnetic controllers will be activated first to bring the satellite from an uncontrolled tumble to a stable Y-axis spinning state. Although the B-dot/Y-spin combination has the advantage of aligning the satellite's Y-axis with the orbit Y-axis faster than the Cross-Product controller, no alignment will be required when precessing the solar panel towards the sun. Investigation showed that the Cross-Product controller requires significantly less torquer on-time to control the satellite's rates. A comparison between the Cross-Product and B-dot/Y-spin controllers can be seen in Table 7.1. The torquer on-time was not only compared during the rate-settling phase, but also during the steady state phase (if the controllers are switched on through telecommands, then the steady state torquer on-time becomes highly significant, since long periods between ground station overpasses might occur). The settling time t_s of the controllers was defined as the time taken from activation until ω_{yo} had reached its reference rate and ω_{xo} and ω_{zo} had settled near zero. The measurement unit for the steady state was defined as the torquer on-time (i.e. the total on-time of the three torquers over a period T) to runtime (i.e. the period T) ratio.

	B-dot/Y-spin	Cross-Product
Settling time t_s (s)	8300	4200
Total torquer on-time for t_s (s)	10952	5145
Settling state torquer on-time ratio	1.320	1.227
Steady state torquer on-time ratio	0.839	0.046

Table 7.1 – A comparison between the B-dot/Y-spin and Cross-Product magnetic controllers

Table 7.1 clearly shows that the Cross-Product is the superior steady state controller, and it was thus

chosen as the preferred controller during de-tumbling. The de-tumbling Cross-Product controller will however require an estimate of ω_{oy}, thus the magnetometer-based RKF (or a Y-axis rate sensor) needs to be activated and have converged before the controller is switched on.

Thereafter the angular momentum vector, which is approximately in the same direction as the main solar panel's normal vector, will slowly be precessed towards the sun using the Precession magnetic controller [15]. This sun-seeking controller will however only be active during the sunlit part of the orbit. The de-tumbling Cross-Product controller will be active during eclipse to "maintain a stable Y-spin rate".

Once sufficient power has been generated, the satellite will enter Nominal Mode when it goes into eclipse. Figure 7.2 shows the systematic flow of the ADCS Safe Mode.

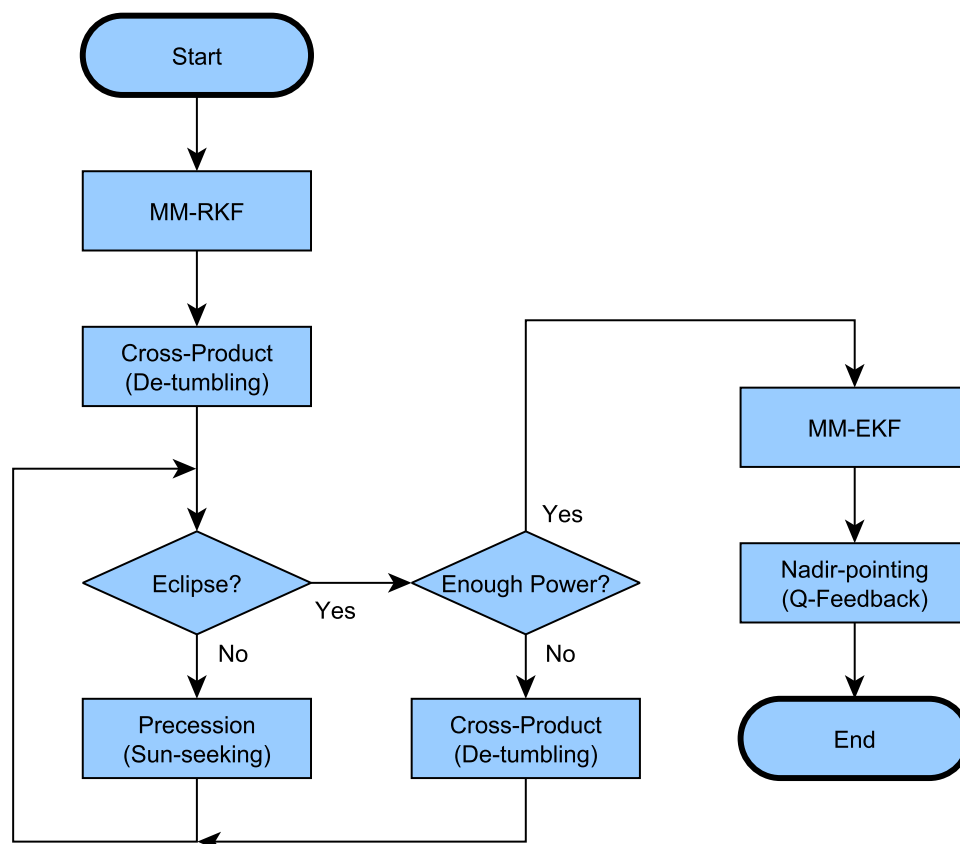
Figure 7.2 – A flow diagram of the ADCS Safe Mode

### 7.4.1   Safe Mode Simulation Results

The Safe Mode simulation results that will be shown in this section will be divided into three parts. The first part will show how the simulated satellite went from an uncontrolled tumble to a reference Y-axis spinning state. The second part will depict the process of precessing the Y-axis momentum

vector towards the sun, which ensures near-maximum power generation. The momentum absorption by the reaction wheels and the transfer to Nominal Mode will finally be shown in the third part.

Reference Y-axis Spin

The initial angular rates (in IRC) for simulating the de-tumbling process were set to a worst case [15] of $\boldsymbol{\omega}_B^I = \begin{bmatrix} 6 & -4 & 10 \end{bmatrix}^T$ °/s. The initial Euler angles were once again arbitrarily set to $\mathbf{RPY} = \begin{bmatrix} -20 & 30 & 10 \end{bmatrix}^T$ °. The magnetometer-based RKF was activated after 200 s and the de-tumbling Cross-Product controller was switched on after 1000 s with a reference rate of $\omega_{yo,ref} = -2°$/s.

Figure 7.3 shows the orbit-referenced angular rates of the satellite during the first part of Safe Mode. ω_{yo} reached its desired value and ω_{xo} and ω_{zo} were near-zero at approximately 5200 s. Figure 7.3 furthermore proves that the de-tumbling controller is relatively robust with respect to initial conditions.
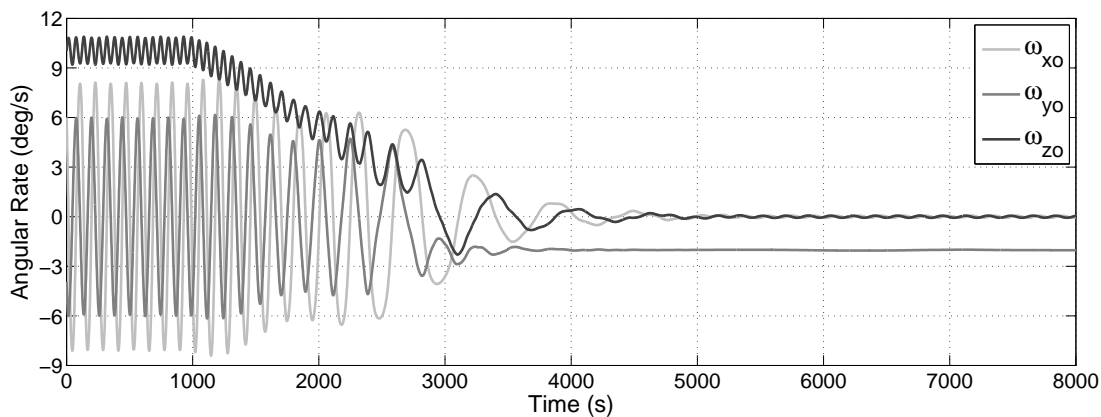
Figure 7.3 – The reference Y-axis spin during Safe Mode

Coarse Sun Tracking

Once the satellite has reached the reference Y-axis spin, the Y-axis angular momentum vector can be precessed towards the sun. This orientation will ensure near-maximum power generation, since the main solar panel will be located on the $-$Y-facet of the satellite.

The precession of an angular momentum vector needs to be executed slowly, otherwise unwanted gyroscopic disturbance torques will be generated. As illustrated in Figure 7.4, the Y-component of the unit sun vector (measured by the CSS) will slowly tend towards -1 over the course of a few orbits. The power generation for the simulation pertaining to Figure 7.4 increased to about 98% (compared to 95% when the B-dot and Y-spin controllers were used during eclipse, as shown in Figure 6.7).

Figure 7.5 shows the orbit-referenced angular rates of the satellite for the simulation that produced Figure 7.4. It can be concluded from these results that the Cross-Product magnetic controller successfully maintained the reference Y-axis spin during the eclipse.
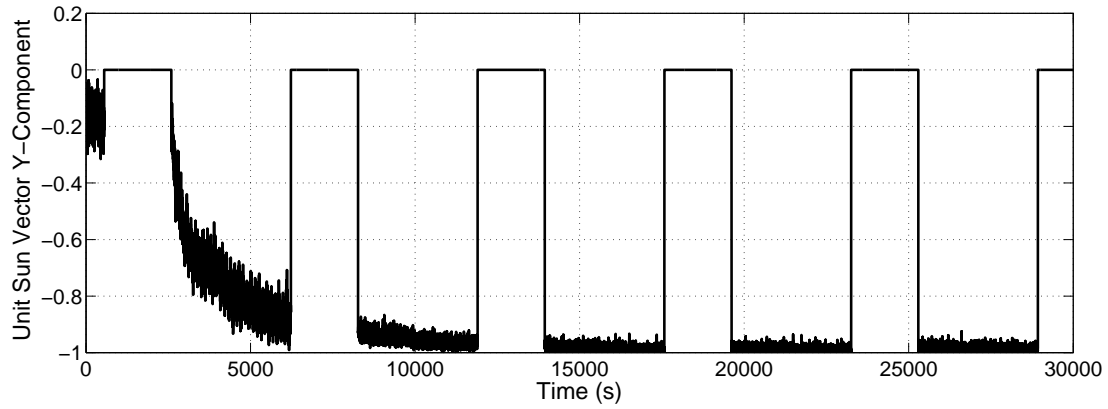
Figure 7.4 – Coarse sun tracking during Safe Mode
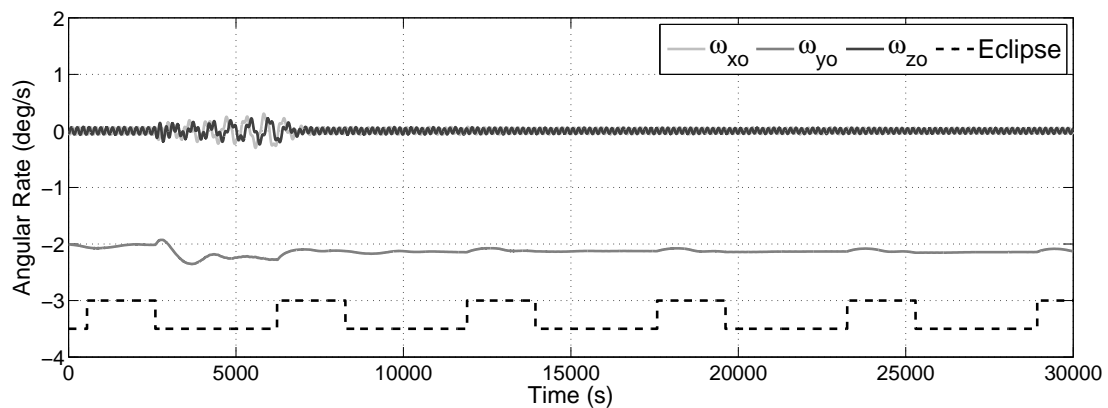
Figure 7.5 – The maintenance of a stable Y-axis spin while performing coarse sun tracking

Transfer To Nominal Mode

With near-maximum power generation during the entire sunlit part of the orbit, the satellite will be able to transfer to Nominal Mode. Once sufficient power has been stored in the batteries, the satellite will switch on the reaction wheels as it goes into eclipse. The Quaternion Feedback wheel controller will be used to stabilise the satellite and to slew to a nadir-pointing attitude, which will complete the transfer to Nominal Mode. Figure 7.6 shows the Euler angles of a simulation of the transfer process, during which the magnetometer-based EKF was used to supply estimates to the wheel controller.
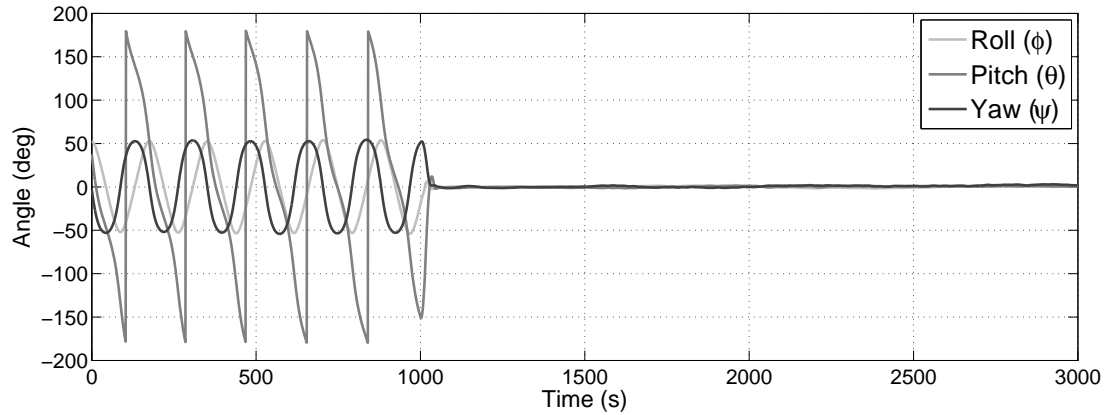
Figure 7.6 – The transfer from Safe Mode to Nominal Mode

7.5   Nominal Mode

Nominal Mode is the default mode in the ADCS state machine. The satellite will initially enter Nominal Mode from Safe Mode once sufficient power has been generated. Nominal Mode will also be entered into when an FMC imaging manoeuvre has been completed or when a ground target being tracked goes out of range. The structure of the Nominal Mode is simple: point nadir and dump angular momentum during eclipse, otherwise track the sun. Figure 7.7 illustrates this basic structure.
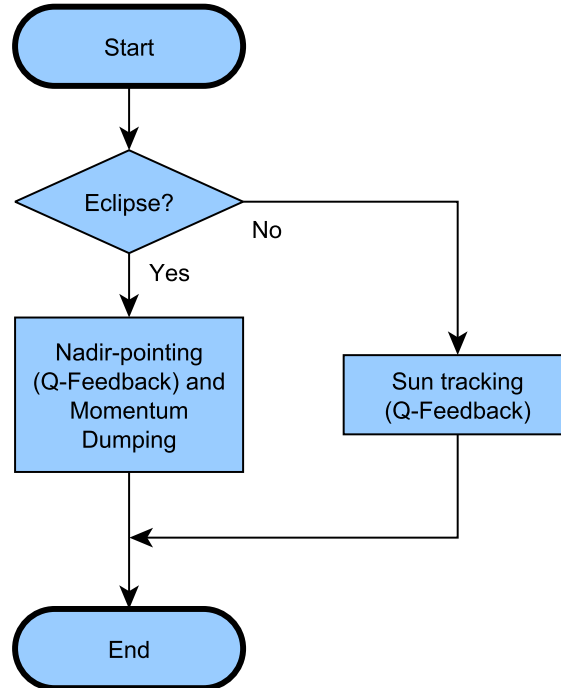
Figure 7.7 – A flow diagram of the ADCS Nominal Mode

The commanded orbit referenced angular rates and vector part of the quaternion are simply all zeros

during nadir-pointing. Since the sun is stationary in the inertial coordinate frame, the commanded $\boldsymbol{\omega}_B^I$ must be zero when performing sun tracking. The attitude command is the orientation where the main solar panel points towards the sun. A technique suggested by Chen et al. [1] can be implemented to calculate the required attitude command. The technique involves defining a unit vector $\mathbf{u}_c$ as

$$\mathbf{u}_c \;=\; \frac{\mathbf{u}_{sp}^{SBC} \times \mathbf{S}_o}{\|\mathbf{u}_{sp}^{SBC} \times \mathbf{S}_o\|} \;,\tag{7.5.1}$$

where $\mathbf{u}_{sp}^{SBC}$ is the unit vector indicating the main solar panel's position and $\mathbf{S}_o$ is the modelled unit sun vector in ORC. The commanded quaternion $\mathbf{q}_c$ can then be defined as

$$\mathbf{q}_c \;=\; \begin{bmatrix} \mathbf{u}_c \sin\frac{\delta}{2} \\[6pt] \cos\frac{\delta}{2} \end{bmatrix} \;,\tag{7.5.2}$$

where δ is the angle between $\mathbf{u}_{sp}^{SBC}$ and $\mathbf{S}_o$. δ can easily be computed using the vector dot-product.

7.5.1 Nominal Mode Simulation Results

Both nadir-pointing and sun tracking were achieved through the Quaternion Feedback wheel controller from Section 6.3.2. The Momentum Dumping magnetic controller from Section 6.2.5 was implemented during eclipse to manage the angular momentum stored in the three wheels. The EKF provided estimates using the magnetometer in eclipse and the FSS in the sunlit part of the orbit.

Figure 7.8 shows typical Euler angles during Nominal Mode. The small disturbances that can be seen in the angles during eclipse were the result of the IGRF and moment of inertia uncertainties.
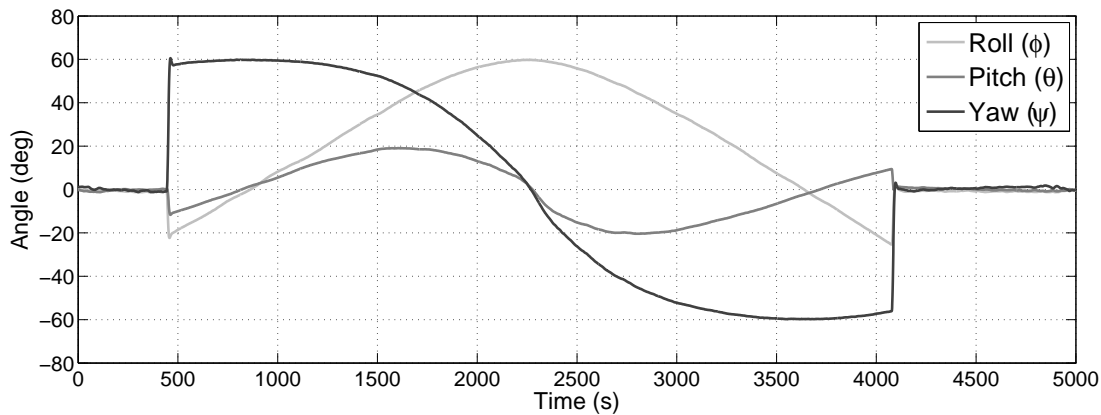
Figure 7.8 – The Euler angles during Nominal Mode

During the Nominal Mode simulation the Y-component of the unit sun vector (measured by the FSS) almost instantly reached -1 when the satellite went into the sunlit part of the orbit, as illustrated in Figure 7.9. Maximum power could thus be generated.
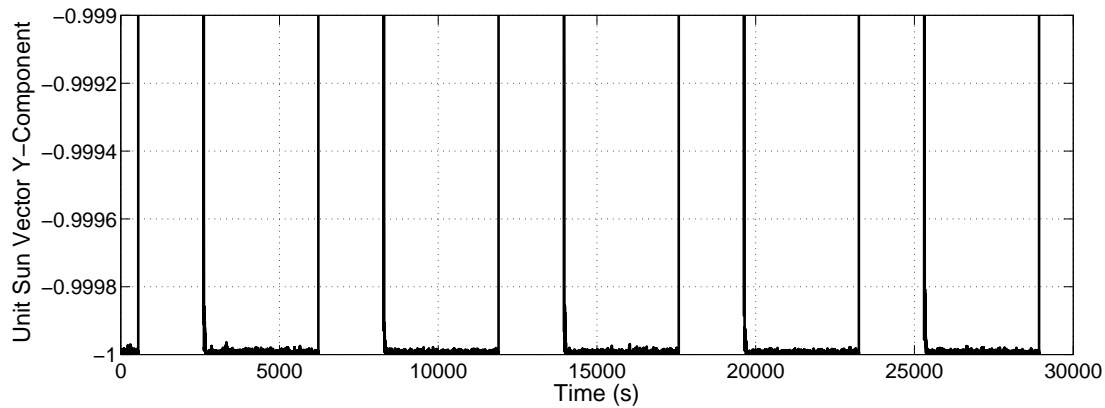
Figure 7.9 – Fine sun tracking during Nominal Mode

The wheel angular momentum for the first few orbits in Nominal Mode can be seen in Figure 7.10. It is clear from these results that the Momentum Dumping magnetic controller was capable of managing the wheel angular momentum.
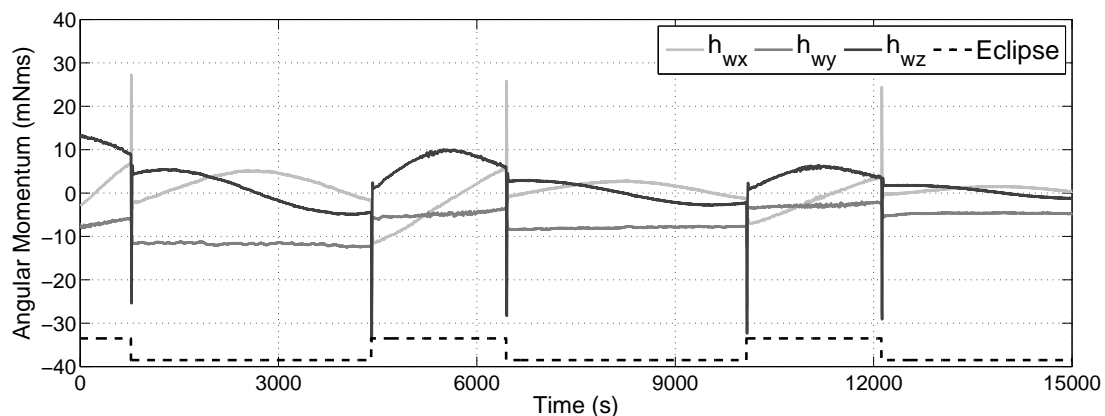
Figure 7.10 – Momentum dumping during Nominal Mode

7.6 FMC Imaging Mode

The most important goal of an EO satellite's ADCS is to provide a stable and accurate platform for imaging manoeuvres. The selection of ADCS hardware and software to be used during imaging greatly determines the image quality. The actuators, controllers, sensors, and estimators that will be used during an imaging manoeuvre must thus be chosen carefully.

An FMC imaging manoeuvre (as depicted in Section 1.3) consists of several stages. The manoeuvre is initialised by pointing in the nadir direction. The satellite will then slew to the roll offset angle and the initial pitch angle, after which the rate about Y_O (ω_Y) will be ramped down to the desired FMC angular rate (ω_{fmc}). Once the angular rate is constant, the imaging of the target area can start. As soon as the imaging has finished, the angular rate will be ramped back up, after which the satellite

will stabilise itself and return to a nadir-pointing attitude. A basic flow diagram of the FMC Imaging Mode can be seen in Figure 7.11.
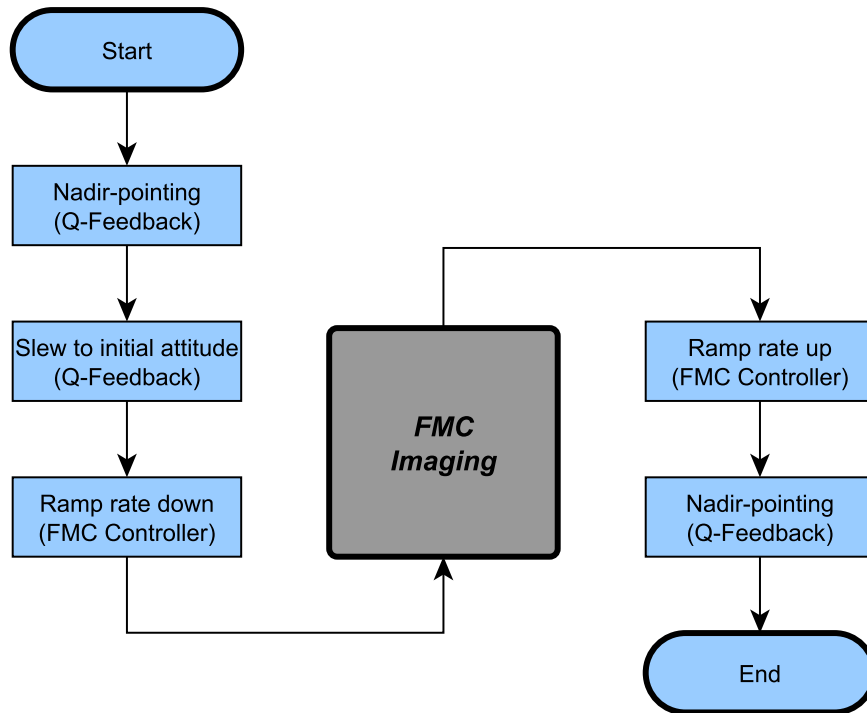
Figure 7.11 – A flow diagram of the ADCS FMC Imaging Mode

7.6.1 FMC Schedule

The scheduling of an imaging manoeuvre is a critical part of any EO satellite mission. The high velocity at which a LEO satellite travels brings forth a tight timing requirement when imaging a target area. An FMC manoeuvre contains added complexity as a result of the rotation compensation, which means that the schedule of this manoeuvre must be planned with precision.

The FMC Imaging Mode will most likely be activated as a result of a telecommand. The telecommand will also contain a specific Unix time t_0 at which the satellite must be at the centre of the imaging manoeuvre (see Section 1.3). The first phase of the FMC Imaging Mode is to slew to nadir-pointing T_{nadir} before t_0 (both in seconds), thus at $t = t_0 - T_{nadir}$.

The next FMC phase, during which the satellite will be slewed to $\mathbf{RPY} = \begin{bmatrix} \phi_{fmc} & \theta_{initial} & 0 \end{bmatrix}^T$, will be executed at $t = t_0 - T_{slew}$. The roll offset angle ϕ_{fmc} will depend on the location of the image target and the initial pitch angle $\theta_{initial}$ will be determined by the duration of the pitch rate ramp T_{ramp} and the pitch angle before imaging θ_{fmc} (see Equation 1.3.32). T_{fmc}, the duration for which the on-board image sensor will be collecting data, can be calculated from L_{image} and n_{fmc} using Equation 1.3.31.

At $t = t_0 - \left(\frac{1}{2}T_{fmc} + T_{buf} + T_{ramp}\right)$ the angular rate about the orbit Y-axis $\boldsymbol{\omega_Y}$ should start being ramped down to a constant rate of $-\boldsymbol{\omega_{fmc}}$. The satellite will be rotating at a rate of $-\omega_{fmc}$ for a period of $T_{fmc} + T_{buf}$, where T_{buf} is an added initial buffer period where the inner reaction wheel speed control dynamics will be allowed to stabilise. A special open loop FMC wheel controller will be used to perform the rate ramp. Once $-\omega_{fmc}$ has been reached at $t = t_0 - \left(\frac{1}{2}T_{fmc} + T_{buf}\right)$, the buffer period will start. The satellite will be ready to start imaging once the buffer period has passed at $t = t_0 - \frac{1}{2}T_{fmc}$.

The imaging payload will stop collecting data at $t = t_0 + \frac{1}{2}T_{fmc}$, after which the angular rate ω_Y will be ramped back up to approximately zero over a period of T_{ramp}. The satellite dynamics will then be allowed to stabilise for a period of T_{stable}, during which the Quaternion Feedback wheel controller will once again be activated.

The FMC imaging manoeuvre will end with a slew to a nadir-pointing attitude at $t = t_0 + \frac{1}{2}T_{fmc} + T_{ramp} + T_{stable}$, after which the FMC Imaging Mode will conclude at $t = t_0 + T_{end}$. The timeline of an FMC imaging manoeuvre (as explained in this section) can be seen in Figure 7.12.
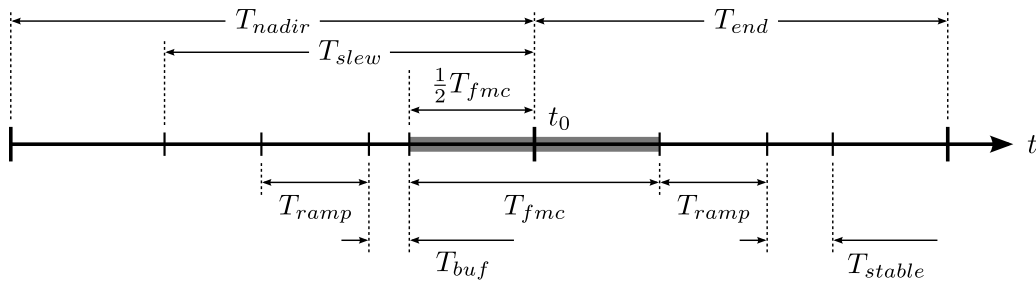
Figure 7.12 – The schedule of an FMC imaging manoeuvre

7.6.2   FMC Wheel Controller

A special wheel controller is required for the FMC imaging manoeuvre. Although the Quaternion Feedback controller offers accurate pointing, its closed loop nature will cause unwanted jitter during imaging. The FMC wheel controller will therefore be an open loop controller.

As a result of the roll offset angle, both the Y-wheel and the Z-wheel will need to be employed to reach the desired Y-axis orbit rate of $\omega_Y = \omega_{fmc}$. The X-wheel will however still be controlled by the Quaternion Feedback controller. The Y-axis and Z-axis open loop control law for the FMC wheel controller is given by

$$N_{wy,req} = I_{yy}N_{fmc}\cos\left(\phi_{fmc}\right) - N_{gyro,y} \tag{7.6.1}$$

$$\text{and} \quad N_{wz,req} = -I_{zz}N_{fmc}\sin\left(\phi_{fmc}\right) - N_{gyro,z} \ , \tag{7.6.2}$$

where N_{fmc} is the torque required to ramp the satellite's rate about Y_O to and from $-\omega_{fmc}$. N_{fmc} is furthermore dependent on T_{ramp} and ω_{fmc} through

$$N_{fmc} \quad = \quad \frac{\omega_{fmc}}{T_{ramp}} \ . \tag{7.6.3}$$

Equation 7.6.3 thus implies that a wheel torque of N_{fmc} is required to ensure a change in angular rate of ω_{fmc} over a period of T_{ramp}. Once the rate ramping process is done, N_{fmc} should be set to zero to ensure a constant FMC rate.

Figure 7.13 illustrates the satellite's orientation during an FMC manoeuvre. It is clear from Figure 7.13 that ω_Y will have components about Y_B and Z_B that will be scaled by $\cos(\phi_{fmc})$ and $\sin(\phi_{fmc})$ respectively, which explains the use of these scalars in Equations 7.6.1 and 7.6.2.
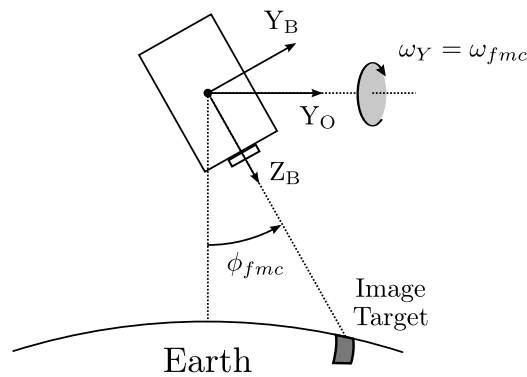
Figure 7.13 – The satellite orientation during an FMC imaging manoeuvre

7.6.3 FMC Implementation

Although the explanation in Chapter 1 builds a basic understanding of an FMC imaging manoeuvre, several factors need to be considered when attempting practical implementation. The rate ramp duration T_{ramp} should firstly be such that the wheel speed commands are not too large, as this will cause large torques and unwanted dynamics. As a rule of thumb for this project, T_{ramp} should be chosen in such a way that $N_{fmc} \approx 0.1 N_{w,max}$, where $N_{w,max}$ is the maximum wheel torque. The duration of T_{ramp} and T_{buf} also influences the required initial pitch angle $\theta_{initial}$, as illustrated by Figure 7.14.

$\theta_{initial}$ can thus be calculated by integrating the area under the curve shown in Figure 7.14, thus

$$\theta_{initial} \quad = \quad \frac{1}{2}\omega_{fmc}T_{ramp} + \omega_{fmc}T_{buf} + \underbrace{\frac{1}{2}\omega_{fmc}T_{fmc}}_{\theta_{fmc}} \ . \tag{7.6.4}$$

The buffer period T_{buf} should furthermore be greater or equal to the wheel speed control loop settling time.
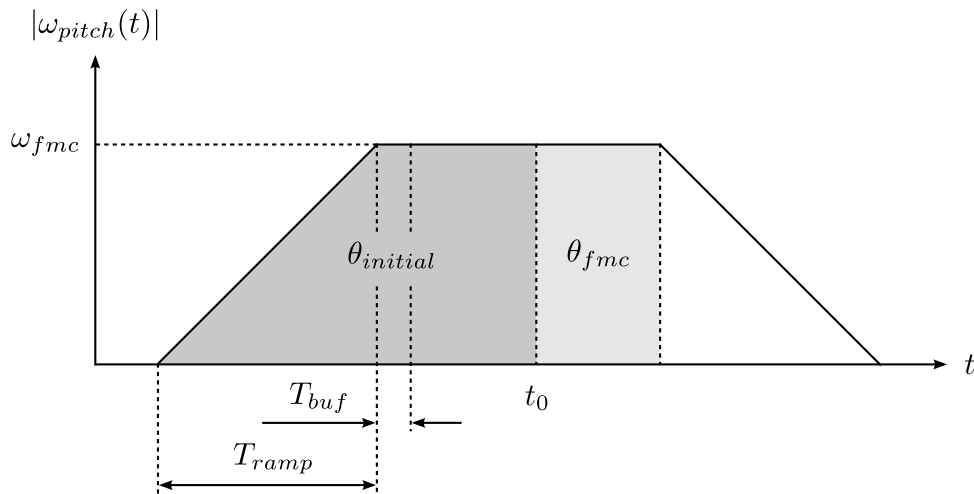
Figure 7.14 – The initial pitch angle for an FMC imaging manoeuvre

7.6.4   FMC Imaging Mode Simulation Results

The correct functioning of the FMC Imaging Mode was verified by simulating the 20 kg EO satellite during an FMC manoeuvre with the following specifications:

- FMC factor (n_{fmc}): 4

- Image length (L_{image}): 40 km

- Altitude (h): $\pm$500 km

- Roll offset angle (ϕ_{fmc}): 30°

- Buffer period (T_{buf}): 5 s

- Ramp duration (T_{ramp}): 20 s

- Start offset time (T_{nadir}): 300 s

- Initial attitude offset time (T_{slew}): 150 s

- Stabilisation period (T_{stable}): 30 s

- Manoeuvre end offset time (T_{end}): 150 s

The FMC imaging duration T_{fmc} and desired angular rate ω_{fmc} were then calculated as approximately 22.6 s and 0.6095°/s (using Equations 1.3.31 and 1.3.30) respectively. Using Equation 7.6.4, the initial pitch angle $\theta_{initial}$ was determined to be 14.53°. The pitch angle at the start of the imaging process was calculated as $\theta_{fmc} = \frac{1}{2}\omega_{fmc}T_{fmc} = 6.90°$.

A star tracker was used to provide measurements to the EKF during the FMC simulation (except during the initial and final slew manoeuvres, where is angular rate will be too high for the star tracker

to produce solutions, and the FSS must be employed). Figure 7.15 shows the Euler angles as produced by the simulation and the resulting orbit-referenced angular rates can be seen in Figure 7.16.
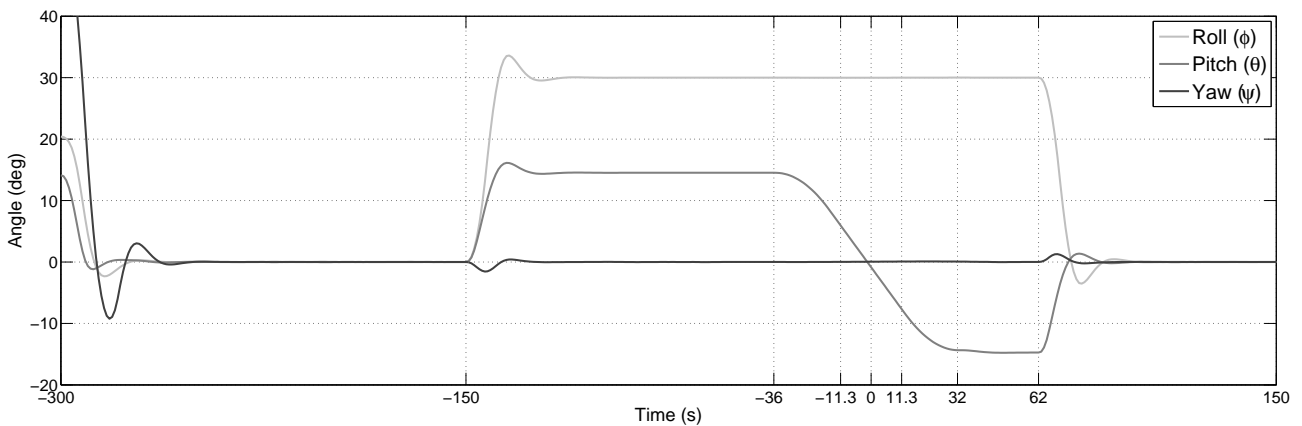
Figure 7.15 – The Euler angles during an FMC imaging manoeuvre
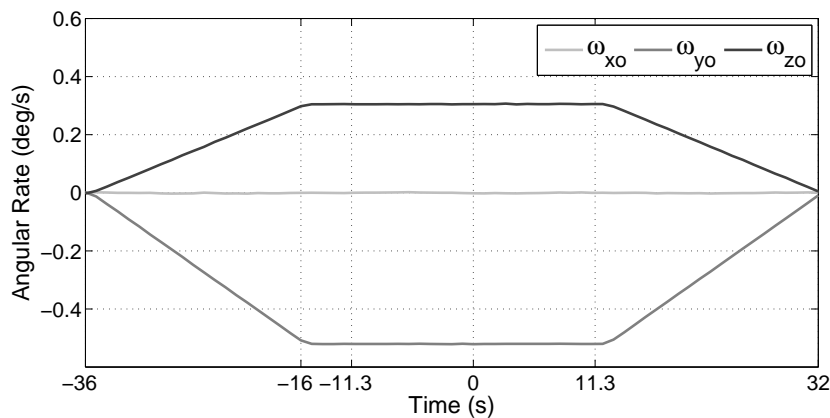
Figure 7.16 – The angular rates during an FMC imaging manoeuvre

The 1-σ roll angle error in Figure 7.15 was calculated as 28.4 arcsec during imaging and the angular rates in Figure 7.16 followed the expected behaviour. The angular rate about Y_O during imaging (which was calculated as $\omega_Y = -\sqrt{\omega_{yo}^2 + \omega_{zo}^2}$) for the same simulation can be seen Figure 7.17. The near-constant offset of roughly 0.007°/s (or 25 arcsec/s) from the desired rate of $\omega_{fmc} = -0.6097°/s$ can be attributed to an initial offset in angular rate (i.e. before the rate was ramped down). The stability of ω_Y, which is defined as the maximum deviation from the mean angular rate of $-0.6031°/s$, was calculated as 6.6 arcsec/s.

7.7   Target Tracking Mode

Precision tracking of targets are often required for satellite missions related to monitoring, communication and Earth observation [1]. A satellite that is to perform target tracking with high accuracy
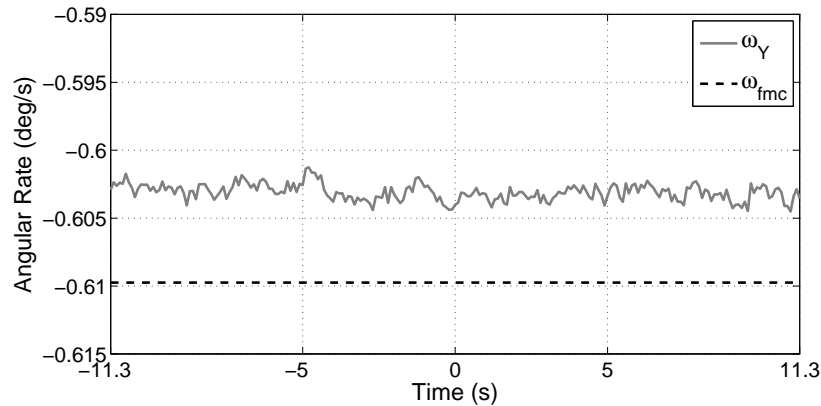
Figure 7.17 – The effective rate about the orbit Y-axis during an FMC imaging manoeuvre

will need to be 3-axis stabilised. Although target tracking can be used for a variety of scenarios, G/S tracking was the only purpose of Target Tracking Mode for the project at hand. It should be noted that the +Z-facet of the satellite was assumed to be the facet that should be pointed towards the target.

As mentioned previously, Target Tracking Mode will most likely be activated from Nominal Mode as soon as a predefined G/S is in range. Knowledge of the satellite position is therefore required by the on-board computer. The method that will be used to calculate the desired attitude and angular rates for tracking a target was presented by Chen et al. in 2000 [1].

The satellite will need to be able to determine whether the target G/S is in range (i.e. in the satellite's field of view of the Earth). The distance D_{max} from the satellite to the furthest point on Earth that the satellite can "see" can be calculated as

$$D_{max} \quad = \quad \sqrt{(R_E + h)^2 - R_E{}^2} \ . \tag{7.7.1}$$

An illustration of the above-mentioned geometry can be seen in Figure 7.18.
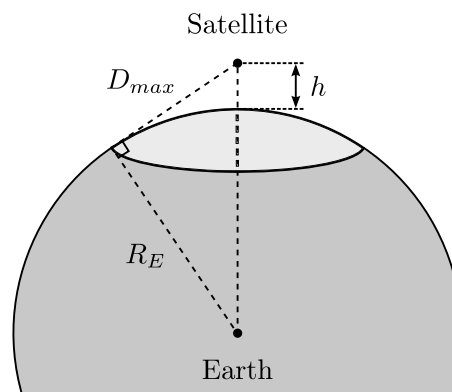
Figure 7.18 – The geometry of the maximum target tracking distance

Once the maximum target distance is known, the current distance to the target must be calculated. The coordinates of a target G/S can be given in geocentric latitude φ (assuming a spherical Earth) and longitude λ. If R_E is the radius of the Earth, then the vector from the centre of the Earth to the target G/S in EFC (or $\mathbf{x}_T^{EFC}$) can easily be calculated as

$$
\mathbf{x}_T^{EFC} \;=\; R_E \begin{bmatrix} \cos(\lambda)\cos(\varphi) \\ \sin(\lambda)\cos(\varphi) \\ \sin(\varphi) \end{bmatrix} . \tag{7.7.2}
$$

Using the transformation matrix $\mathbf{A}_{EFC}^{EIC}$ (see Section 1.3), $\mathbf{x}_T^{EFC}$ can be transformed to $\mathbf{x}_T^{EIC}$, a vector in EIC, thus

$$
\mathbf{x}_T^{EIC} \;=\; \mathbf{A}_{EFC}^{EIC}\mathbf{x}_T^{EFC} . \tag{7.7.3}
$$

$\mathbf{x}_T^{EIC}$ and $\mathbf{r}_{sat}$ (the satellite's position vector in EIC, as produced by the SGP4) can then be used to calculate $\mathbf{x}_{S/T}^{ORC}$, the vector from the satellite to the target G/S in ORC, using

$$
\mathbf{x}_{S/T}^{ORC} \;=\; \mathbf{A}_{EIC}^{ORC}\left(\mathbf{x}_T^{EIC} - \mathbf{r}_{sat}\right) , \tag{7.7.4}
$$

where $\mathbf{A}_{EIC}^{ORC}$ is the transformation matrix from EIC to ORC as defined by Equation 1.3.5. If the magnitude of $\mathbf{x}_{S/T}^{ORC}$ is larger than D_{max}, then the target is not in range and the satellite will continue operating in Nominal Mode. However, if $\|\mathbf{x}_{S/T}^{ORC}\| < D_{max}$, Target Tracking Mode will be activated.

Only the direction of $\mathbf{x}_{S/T}^{ORC}$ is of use to the satellite's ADCS, hence the unit vector $\mathbf{u}_{S/T}^{ORC}$ must be calculated by normalised $\mathbf{x}_{S/T}^{ORC}$, thus

$$
\mathbf{u}_{S/T}^{ORC} \;=\; \frac{\mathbf{x}_{S/T}^{ORC}}{\|\mathbf{x}_{S/T}^{ORC}\|} . \tag{7.7.5}
$$

Once $\mathbf{u}_{S/T}^{ORC}$ is known, a similar process as the Nominal Mode sun tracking from Section 7.5 can be followed to calculate the quaternion error vector $\mathbf{q}_e$. In this case however, the unit vector $\mathbf{u}_c$ from Equation 7.5.1 must be calculated using the satellite body Z-axis unit vector $\mathbf{z}_b$ as

$$
\mathbf{u}_c \;=\; \frac{\mathbf{z}_b \times \mathbf{u}_{S/T}^{ORC}}{\|\mathbf{z}_b \times \mathbf{u}_{S/T}^{ORC}\|} . \tag{7.7.6}
$$

The commanded quaternion can be determined using Equation 7.5.2, after which $\hat{\mathbf{q}}_e$ can be calculated using Equations 1.3.21 and 1.3.22.

The satellite's angular rate for target tracking will not be zero (as was the case with the Nominal Mode sun tracking), since the target G/S is fixed with respect to the Earth and will thus be rotating relative to the orbiting satellite. The commanded angular rate $\boldsymbol{\omega}_c^{ORC}$ can be calculated using

$$
\boldsymbol{\omega}_c^{ORC} \;=\; \mathbf{u}_{S/T}^{ORC} \times \dot{\mathbf{u}}_{S/T}^{ORC} , \tag{7.7.7}
$$

where $\dot{\mathbf{u}}_{S/T}^{ORC}$ is the time derivative of $\mathbf{u}_{S/T}^{ORC}$. Although the calculation of $\dot{\mathbf{u}}_{S/T}^{ORC}$ is quite extensive, the time derivative can also be approximated as

$$(\dot{\mathbf{u}}_{S/T}^{ORC})_k \quad \approx \quad \frac{1}{T_s}\left((\mathbf{u}_{S/T}^{ORC})_k - (\mathbf{u}_{S/T}^{ORC})_{k-1}\right) \; , \tag{7.7.8}$$

where T_s is the wheel control sample period [3]. Since $\mathbf{u}_{S/T}^{ORC}$ is aligned with the body Z-axis, the commanded Z-axis angular rate will be zero. Equation 7.7.7 thus reduces to

$$\boldsymbol{\omega}_c^{ORC} \quad = \quad \begin{bmatrix} -\dfrac{\dot{u}_y}{u_z} \\[2mm] \dfrac{\dot{u}_x}{u_z} \\[2mm] 0 \end{bmatrix} \; , \tag{7.7.9}$$

where u_z, $\dot{u}_x$, and $\dot{u}_y$ are the respective components of $\mathbf{u}_{S/T}^{ORC}$ and $\dot{\mathbf{u}}_{S/T}^{ORC}$. Once the ORC angular rate error $\hat{\boldsymbol{\omega}}_e$ has been calculated using

$$\hat{\boldsymbol{\omega}}_e \quad = \quad \hat{\boldsymbol{\omega}}_B^O - \boldsymbol{\omega}_c^{ORC} \; , \tag{7.7.10}$$

the Quaternion Feedback controller from Chapter 6 can be used to track the target G/S.

7.7.1   Target Tracking Mode Simulation Results

The ground station at the ESL at Stellenbosch University was chosen as the target for the Target Tracking Mode simulation. The latitude and longitude of this G/S is known to be $-33.928290°$ and $18.866329°$ respectively. The maximum target distance from an altitude of ± 500 km was calculated as $D_{max} = 2574.5$ km. Since G/S tracking can occur at any time during the orbit, the magnetometer-based EKF was chosen as the estimator for Target Tracking Mode. One overpass of 681 s was simulated, and the resulting Euler angles can be seen in Figure 7.19.
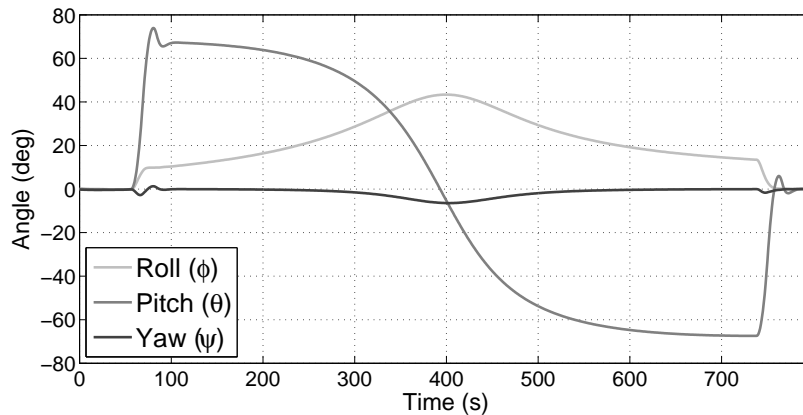
Figure 7.19 – The Euler angles during the tracking of a ground station

7.8   Pointing Accuracy and Stability

The required pointing accuracy and stability of the satellite were determined in Chapter 3 as 825 arcsec and 73 arcsec/s respectively. The actual performance of the satellite's ADCS with respect to pointing and stability needed to be determined to verify that the the given requirements were met.

The pointing accuracy of the satellite was determined by performing a target tracking manoeuvre using star tracker measurements. The resulting pointing error is shown in Figure 7.20. The maximum pointing error of 63 arcsec that can be seen in Figure 7.20 is well below the required 825 arcsec.
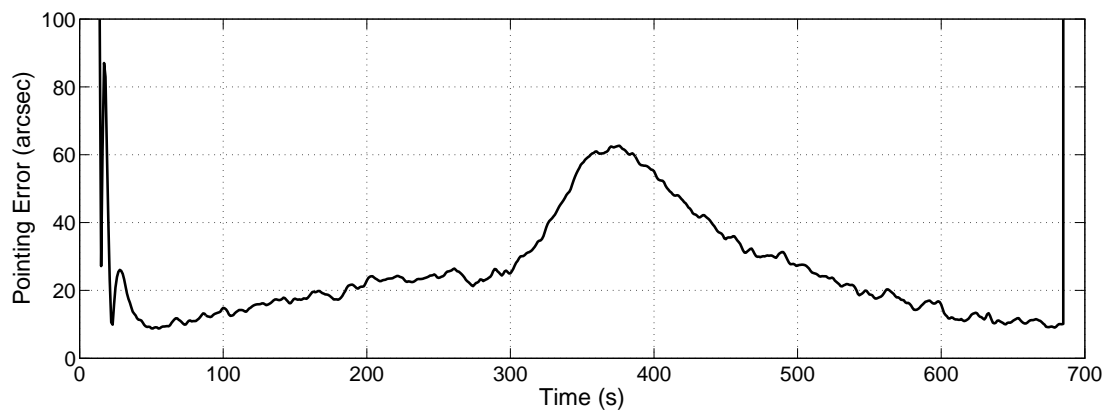
Figure 7.20 – The pointing error of the 20 kg satellite

The stability of the satellite during imaging was be determined from Figure 7.17 as 6.6 arcsec/s, which is an order of magnitude smaller than the required stability of 72.8 arcsec/s. The achieved stability can be translated to a maximum pixel smear of only 1% (compared to the required maximum smear of 10%).

7.9   Summary

The five modes of the ADCS state machine were broadly covered in this chapter. An overview of each mode (except for Idle Mode) was firstly given, after which the implementation of the mode was discussed. The correct functioning of the relevant mode was verified through simulation at the end of each section.

A Safe Mode was firstly defined to serve as both the de-tumbling mode and the fail-safe mode. The simulation results showed that the satellite will be able to enter a Y-spin within one orbit, even when worst case initial tumbling rates were assumed. It was furthermore found that the satellite would be able to perform coarse sun tracking within a few orbits after reaching the Y-spin, which would increase power generation to 98%.

The Nominal Mode was chosen as the default ADCS mode, during which fine sun tracking will be performed in the sunlit part of the orbit. The momentum dumping capability of the magnetic torquers during eclipse (when the satellite will be in a nadir-pointing attitude) was also proved.

The satellite will activate its primary payload during the FMC Imaging Mode. A timing schedule and an open loop wheel controller was furthermore defined for the FMC manoeuvres. The simulation results showed that the satellite will be able to perform an FMC factor 4 manoeuvre while maintaining a roll offset angle with a 1-σ error of 28.4 arcsec. The MoI uncertainty proved to affect pitch rate of the satellite during imaging only slightly.

A Target Tracking Mode was also included in the state machine to provide the satellite with the ability to track a predefined target G/S. This mode was furthermore chosen to have a lower priority than the FMC Imaging Mode. The simulation results indicated that the satellite will be able to successfully track G/S during the Target Tracking Mode.

It was lastly established through simulation that the satellite will be able to perform a tracking manoeuvre with a pointing error below 70 arcsec and stability less than 7 arcsec/s, both of which are well within the given requirements.

Chapter 8

Conclusions and Recommendations

8.1 Research Summary and Conclusions

This project was aimed at the development of an attitude determination and control system for a 20 kg Earth observation satellite.

The pre-flight performance of an ADCS must be determined through simulation. A realistic simulation environment was thus created for this purpose using MATLAB and Simulink. The simulation environment included various space environment models, such as an SGP4 orbit propagator, a 10th order geomagnetic field model based on the IRGF coefficients, and a sun position model. Three disturbance models were also included in the simulation environment, namely gravity gradient, aerodynamic disturbance, and wheel imbalance. The satellite's dynamics and kinematics were simulated using Simulink's second order solver for continuous integration.

An initial ADCS analysis was performed based on a given set of requirements and specifications for the satellite and its ADCS. A simple position and pointing budget was compiled and it was indicated that a high accuracy sensor was needed to meet the require maximum attitude knowledge error of 118 arcsec. Investigation showed that a nano-satellite star tracker would be able to provide the required accuracy. The remaining sensors that were chosen for the ADCS hardware suite included a magnetometer, a coarse sun sensor, and a fine sun sensor. The satellite requirements were furthermore translated to minimum actuator requirements. It was found that the satellite would be able to satisfy both the agility and stability requirements using three reaction wheels (even in the presence of absolute worst case disturbances). The minimum required angular momentum storage and torque generation specifications for the wheels were calculated as 37 mNms and 1.1 mNm respectively, which can easily be satisfied using commercially available products. An example commercial reaction wheel was used to determine that the maximum angular rate jitter would remain well within the stability requirement. Magnetic torquers that could generate at least 0.64 Am^2 were furthermore required to dump the wheels' angular momentum.

Mathematical models for the chosen sensors and actuators were then derived and implemented in the simulation environment. The placement of the fine sun sensor and the star tracker was defined in such a way that when the satellite is in a nadir-pointing attitude, the fine sun sensor would be on the sunlit part of the orbit plane and the star tracker would be on the opposite side of the satellite. The star tracker was furthermore placed at 45° between the +Y-axis and the −Z-axis, which would ensure valid measurements during an entire imaging manoeuvre. The measurement noise for all the sensors were approximated as zero-mean Gaussian, of which the standard deviation was based on a typical commercial version of the relevant sensor. The torquers were modelled as being affected by quantisation only, whereas a complex reaction wheel model was derived using a practical example based on a PID controller.

Once the hardware models were in place, the ADCS software needed to be developed. Several estimators were investigated, namely TRIAD and Optimized TRIAD (which are both geometeric attitude determination methods), a rate Kalman Filter, and a 7-state Extended Kalman Filter. The Optimized TRIAD method was firstly found to provide increased accuracy compared to the TRIAD method. The magnetometer-based RKF proved to be a useful estimator during de-tumbling, since only rough rate estimates would be required. The EKF on the other hand was found to provide accurate estimates of the attitude quaternion and the body angular rates, with 1-σ errors below 25 arcsec and 5 arcsec/s respectively when using star tracker measurements.

Five magnetic controllers were investigated during the course of this project. Simulation results indicated that the B-dot controller was successfully able to align the body Y-axis with the orbit Y-axis and that the Y-spin controller could thereafter bring the satellite to a reference Y-spin rate. The Cross-Product magnetic controller was successfully implemented as both a de-tumbling controller and a Y-wheel momentum management controller. Coarse sun tracking was demonstrated through the Precession controller, which has the ability to precess the body angular momentum vector towards the sun. A 3-axis momentum dumping magnetic controller was lastly discussed and the working thereof was verified through simulation.

Simulation showed that the Y-momentum wheel controller was able to absorb the Y-spin body angular momentum, after which a reference pitch angle was adequately followed. The globally stable Quaternion Feedback controller was utilised to perform 3-axis reaction wheel control. A desired attitude could be reached within tens of seconds with 1-σ errors (on each of the Euler angles) less than 45 arcsec. The effect of wheel speed measurement noise on the pointing accuracy and stability of the satellite was also investigated and it was concluded that the filtering property of the EKF rejected the wheel speed noise to a great extent.

With the hardware and numerous possible software controllers defined, the ADCS state machine (or execution strategy) could be constructed. A Safe Mode was defined during which the satellite will

be de-tumbled, be it after release or after a serious system error. The de-tumbling process involves bringing the satellite's Y-axis body rate to a reference value (which was chosen as $-2°/s$) and then precessing the Y-axis momentum vector (which is in the same direction as the main solar panel's outward normal vector) towards the sun, i.e. coarse sun tracking. The power generation increased to approximately 98% after 3 orbits. The Cross-Product was preferred over the B-dot/Y-spin combination to enforce the Y-spin during de-tumbling, mainly because the latter required unnecessary control effort to align the body Y-axis with the orbit Y-axis. The magnetic Precession controller was used to perform the sun tracking during the sunlit part of the orbit. The Y-momentum wheel controller was not considered for this mode, because it was deemed Simulation results indicated that the Safe Mode was able to successfully de-tumble the satellite from worst case initial conditions.

The next state of the ADCS state machine was chosen to be Nominal Mode, which is the default mode when not performing imaging or target tracking manoeuvres. It was shown that the satellite could easily enter this mode from Safe Mode using a magnetometer-based EKF and the Quaternion Feedback wheel controller. The last-mentioned process will however only be executed once it has been established that sufficient power has been generated. When in Nominal Mode, the satellite will perform fine sun tracking in the sunlit part of the orbit and point nadir during eclipse. Simulations also proved that the magnetic Momentum Dumping controller could successfully manage the reaction wheel angular momentum during eclipse.

A Target Tracking Mode was also added to the state machine to provide the satellite with the ability to target a predefined ground station. The simulation results proved that the satellite was able to track the target for the entire duration of an overpass.

The imaging payload, which was chosen to point in the +Z-direction, will be employed with the assistance of an FMC Imaging Mode, which enjoys the highest priority in the ADCS state machine. Simulation results indicated that the satellite could successfully perform FMC factor 4 with a $30°$ roll offset angle using a star-tracker-based EKF and a 1-σ roll angle error of less than 30 arcsec was achieved during imaging.

The pointing accuracy of the satellite was determined through a star-tracker-based target tracking manoeuvre to be less than 70 arcsec, which is a fraction of the required pointing accuracy of 825 arcsec. The stability of the satellite during imaging was calculated as approximately 7 arcsec/s, resulting in only a 1% image pixel smear (compared to the desired maximum smear of 10%).

In conclusion, this project can be regarded as being successfully undertaken. A realistic simulation environment, containing space environment and ADCS hardware models, was created to serve as a platform for analysing the performance of an ADCS. Various estimators and controllers were invest-igated and based on the results thereof, a comprehensive ADCS state machine was developed for a 20

kg Earth observation satellite which met the required specifications with ease.

8.2   Recommendations and Improvements

No research project is without flaw. Although a comprehensive effort was made to ensure that every aspect of this project was executed thoroughly, room for improvement will inevitably exist. This section will be dedicated to identifying areas of the project where improvement is possible and to recommend alternative or additional solutions.

8.2.1   Orbit Propagation Error

The effect of orbit propagation errors was not modelled in this project. The satellite position vector propagated by the SGP4 is known to be slightly erroneous, which means that the modelled geomagnetic field and sun position vectors will also contain errors when based on the SGP4. The simulation environment can be adapted to account for the error by adding noise to the position vector on which the satellite's on-board environment models are based. The original position vector should then be used to calculate the true models. This process will result in a more realistic simulation platform. In practice, a GPS can be added to the ADCS hardware suite to minimise the above-mentioned errors.

8.2.2   ADCS Hardware

This project did not include any form of hardware redundancy. As discussed in Chapter 1, the use of redundant ADCS hardware comes at the cost of increased satellite mass, volume, cost, and power requirements. If one of the last-mentioned constraints can in any way be sacrificed, adding redundancy on mission critical components might increase the lifetime of the satellite significantly.

A redundant **fourth reaction wheel** can be added to the current system to ensure stability and fine pointing even in the event of one of the four wheels failing. Either a pyramid or a tetrahedral configuration can be used to align the wheels. Another advantage of adding a fourth wheel is that the maximum angular momentum and torque that can be stored and generated per axis will increase. The four wheels will run at a bias wheel speed (resulting in zero total angular momentum), which will minimise the effect of the motor non-linearities near zero wheel speed. An inner angular momentum management controller will furthermore be required if four reaction wheels are used. The controller will ensure that that the norm of the angular momentum stored in the wheels remains at a minimum.

A **second star tracker** can be added to the current ADCS hardware suite to ensure redundancy with regards to accurate position knowledge. The additional star tracker can also be used in conjunction with the first star tracker to increase the accuracy of attitude estimates during imaging. A **Y-axis rate sensor**, for example a micro-electromechanical systems (MEMS) rate sensor, can also be added to provide body Y-axis rate measurements during Safe Mode, thus replacing the magnetometer-based

RKF. The rate sensor will not necessarily improve the current system, but it will rather add some form of redundancy.

8.2.3 Solar Panels

Throughout the project it was assumed that a main solar panel on the $-Y$-facet of the satellite will provide sufficient power throughout the entire orbit. This might not be the true if the power consumption of the various modules of the satellite is too high, in which case deployable solar panels can be added. It is also possible that the single Y-axis solar panel will not generate enough power to de-tumble the satellite, since sun tracking will not yet be possible at that stage. One solution might be to add smaller solar panels on both the X-facets. Another solution might be to place solar panels on both of the X-facets and both of the Y-facets and to perform a gravity-gradient-stabilised Z-spin during de-tumbling (i.e. "barbecue" mode).

8.2.4 Wheel Control During Eclipse

Since power is limited during eclipse, every effort should be made to minimise control effort. The gains of the Quaternion Feedback controller that enforces a stable nadir-pointing attitude can firstly be scaled down. Furthermore, the yaw angle and Z-axis body rate do not need to be controlled when the satellite is in a nadir-pointing attitude. Both of the above-mentioned strategies should lower the total control energy during eclipse.

8.2.5 Angular Momentum Management

For the project at hand it was chosen that angular momentum dumping (using the magnetic torquers) will only occur during eclipse. It might however be that the power system of the satellite will not be able to supply a sufficient amount of power during eclipse to perform momentum dumping. In such a case, momentum dumping can also be performed during the sunlit part of the orbit, but only while FMC imaging is not being executed. As a result, the worst case jitter caused by wheel imbalance will also be less than before, which might improve the image quality.

8.2.6 Wheel Speed Controller

The PID wheel speed controller discussed in Chapter 4 caused overshoot to occur when a speed step command was given. Although the commanded change in angular momentum (which is proportional to the commanded change in wheel speed) is broken up into 10 smaller steps (see Chapter 6) to lessen the effect of the overshoot, a more elegant approach would be to design a state space wheel speed controller that would not cause any overshoot. To verify the last-mentioned statement, integral state space control was implemented on the engineering model RW from Chapter 4. The result of a practical test, which can be seen in Figure 8.1, confirms that the state space controller offers an acceptable solution to the overshoot of the PID controller.
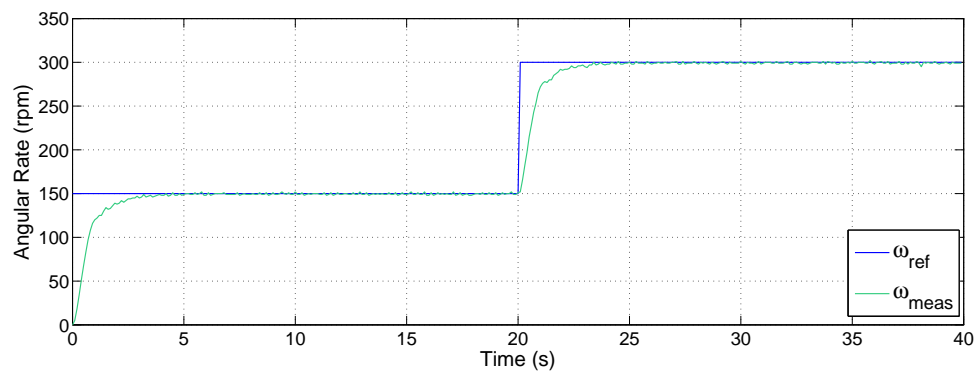
Figure 8.1 – The performance of a state space wheel speed controller

Bibliography

[1]   Chen, X., Steyn, W.H. and Hashida, Y.: Ground-Target Tracking Control of Earth-Pointing Satellites. In: *Proceedings AIAA-2000-4547, Guidance, Navigation, and Control Conference and Exhibit*, pp. 1–11. AIAA, 2000.

[2]   Wertz, J.R. (ed.): *Spacecraft Attitude Determination and Control*. Kluwer Academic Publishers, 1978.

[3]   Steyn, W.H.: AODCS Wheel Control. Notes.

[4]   Hoots, F.R. and Roerich, R.L.: SPACETRACK REPORT NO. 3. December 1980.

[5]   Vallado, D.A., Crawford, P., Hujsak, R. and Kelso, T.S.: Revisiting Spacetrack Report #3. Tech. Rep., American Institute of Aeronautics and Astronautics, 2006. AIAA 2006-6753.

[6]   Finlay, C.: International Geomagnetic Reference Field, IAGA. Online, January 2010. Accessed on 27/11/2014.
      Available at: `http://www.ngdc.noaa.gov/IAGA/vmod/igrf.html`

[7]   Vallado, D.A.: *Fundamentals of Astrodynamics and Applications*. 2nd edn. Microcosm Press, Kluwer Academic Publishers, 2001.

[8]   Wertz, J.R. and Larson, W.J. (eds.): *Space Mission Analysis and Design*. 3rd edn. Microcosm Press, Kluwer Academic Publishers, 1999.

[9]   Wertz, J.R., Everett, D.F. and Puschell, J.J. (eds.): *Space Mission Engineering: The New SMAD*. Microcosm Press, 2011.

[10] Steyn, W.H. and Kearney, M.-A.: An Attitude Control System for ZA-AeroSat subject to significant Aerodynamic Disturbances. *19th World Congress of the The International Federation of Automatic Control*, August 2014. IFAC2014.

[11] Steyn, W.H. and Lappas, V.: Cubesat solar sail 3-axis stabilization using panel translation and magnetic torquing. *Aerospace Science and Technology*, vol. 15, no. 6, September 2011.

[12] Bialke, B.: A Compilation of Reaction Wheel Induced Spacecraft Disturbances. *20th Annual AAS Guidance and Control Conference*, February 1997.

[13] Bialke, B.: High Fidelity Mathematical Modeling of Reaction Wheel Performance. *21st Annual AAS Guidance and Control Conference*, February 1998.

[14] Erlank, A.O.: *Development of CubeStar, A CubeSat-Compatible Star Tracker.* Master's thesis, Stellenbosch University, 2013.

[15] Steyn, W.H.: An Attitude Control System for SumbandilaSAT, an Earth Observation Satellite. *The 4S Symposium, ESA Proceedings*, May 2008.

[16] Steyn, W.H.: *A Multi-Mode Attitude Determination and Control System for Small Satellites.* PhD Dissertation, Stellenbosch University, 1995.

[17] Black, H.: A Passive System for Determining the Attitude of a Satellite. *AIAA Journal*, vol. 2, no. 7, pp. 1350 – 1351, 1964.

[18] Bar-Itzhack, I.Y. and Harman, R.R.: Optimized TRIAD Algorithm for Attitude Determination. *Journal of Guidance, Control and Dynamics*, vol. 20, no. 1, pp. 208 – 211, 1997.

[19] Kalman, R.E.: A New Approach to Linear Filtering and Prediction Problems. *Journal of Fluids Engineering*, vol. 82, pp. 35 – 45, March 1960.

[20] Kalman, R.E. and Bucy, R.S.: New Results in Linear Filtering and Prediction Theory. *Journal of Fluids Engineering*, vol. 83, pp. 95 – 108, March 1961.

[21] Steyn, W.H.: AODCS EKF Estimation, November 2004. Notes.

[22] Martel, F., Pal, P.K. and Psiaki, M.: Active Magnetic Control System for Gravity Gradient Stabilized Spacecraft. *Proceedings of the 2nd Annual AIAA/USU Conference on Small Satellites*, September 1988. Utah State University.

[23] Hodgart, M.S. and Ong, W.T.: Attitude Determination and Control for Low Earth Orbit Micro-satellites. *IEEE/SAIEE Symposium on Small Satellites and Control Systems*, October 1994.

[24] Steyn, W.H. and Hashida, Y.: An Attitude Control System for a Low-Cost Earth Observation Satellite with Orbit Maintenance Capability. *13th Annual AIAA/USU Conference on Small Satellites*, August 1999.

[25] Auret, J.: *Design of an Aerodynamic Attitude Control System for a CubeSat.* Master's thesis, Stellenbosch University, March 2012.

[26] Thomson, W.T.: Spin Stabilisation of Attitude against Gravity Torque. *Journal of Astronautical Science*, , no. 9, pp. 31 – 33, 1962.

[27] Steyn, W.H.: AODCS Momentum Dumping, 2004. Notes.

[28] Wie, B., Weiss, H. and Arapostathis, A.: Quaternion Feedback Regulator for Spacecraft Eigenaxis Rotations. *Journal of Guidance, Control, and Dynamics*, vol. 12, no. 3, May - June 1989.

Appendix A

Small Satellite Database

The following small satellites were included in the ADCS hardware investigation discussed in Chapter 1 (ordered by launch date, earliest first):

- Tubsat A
- Tubsat B
- MuSat 1
- TechSat
- MightySat I
- SUNSAT
- OPAL
- SNAP-1
- UniSat 1
- SAPPHIRE
- Proba-1
- MAROC-Tubsat
- FedSat
- UniSat 2

- MIMOSA
- UniSat 3
- UniSat 4
- LAPAN-Tubsat
- FalconSat-3
- CanX-6
- SORUNSAT 1
- SpriteSat
- Nanosat 1B
- SumbandilaSat
- EduSat
- SRMSAT
- VesselSat-1
- VesselSat-2

- ALMASat-1
- HORYU-2
- SDS-4
- UniSat 5
- ShindaiSat
- TeikyoSat 3
- SOCRATES
- UNIFORM 1
- BugSat-1
- UniSat 6
- AISat 1
- CanX-4/5
- Hodoyoshi 1
- Tsubame

Appendix B

Extended Kalman Filter Derivations

System Perturbation Jacobian Matrix

This section will show the derivation of the Jacobian matrix $\mathbf{F}_t$ that must be constructed in the execution of the full state EKF, as described in Section 5.4.

The continuous system perturbation matrix $\mathbf{F}_t$ can be constructed by determining its individual components, thus

$$
\mathbf{F}_t \;\; = \;\; \begin{bmatrix} \frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \boldsymbol{\omega}_B^I} & \frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \mathbf{q}} \\[2mm] \frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}_B^I} & \frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}} \end{bmatrix}_{\boldsymbol{\omega}_B^I = \hat{\boldsymbol{\omega}}_B^I, \; \mathbf{q} = \hat{\mathbf{q}}} . \tag{B.1}
$$

Note that the subscript 't' indicating the time domain has been dropped from Equation 5.4.5 to simplify the derivation. The non-linear function $\boldsymbol{f}(\mathbf{x})$ can be separated into two parts: a non-linear function describing $\dot{\boldsymbol{\omega}}_B^I$ and a non-linear function describing $\dot{\mathbf{q}}$. The continuous non-linear system equation with regards to $\dot{\boldsymbol{\omega}}_B^I$ is the Euler dynamic equation, or

$$
\dot{\boldsymbol{\omega}}_B^I \;\; = \;\; \mathbf{J}^{-1}\Big(\mathbf{N}_c + \mathbf{N}_d - \boldsymbol{\omega}_B^I \times \Big(\mathbf{J}\boldsymbol{\omega}_B^I + \mathbf{h}_w\Big)\Big) . \tag{B.2}
$$

The individual components of Equation B.2 can also be expressed as

$$
\dot{\omega}_{xi} \;\; = \;\; \frac{1}{I_{xx}} \left(N_{cx} + N_{dx} - \omega_{yi}(I_{zz}\omega_{zi} + h_z) + \omega_{zi}(I_{yy}\omega_{yi} + h_y)\right) \tag{B.3}
$$

$$
\dot{\omega}_{yi} \;\; = \;\; \frac{1}{I_{yy}} \left(N_{cy} + N_{dy} - \omega_{zi}(I_{xx}\omega_{xi} + h_x) + \omega_{xi}(I_{zz}\omega_{zi} + h_z)\right) \tag{B.4}
$$

$$
\dot{\omega}_{zi} \;\; = \;\; \frac{1}{I_{zz}} \left(N_{cz} + N_{dz} - \omega_{xi}(I_{yy}\omega_{yi} + h_y) + \omega_{yi}(I_{xx}\omega_{xi} + h_x)\right) . \tag{B.5}
$$

Using Equations B.3 to B.5, $\frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \boldsymbol{\omega}_B^I}$ can be determined by taking each individual partial derivative, which delivers

$$\frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \boldsymbol{\omega}_B^I} = \begin{bmatrix} 0 & \frac{\omega_{zi}(I_{yy}-I_{zz})-h_z}{I_{xx}} & \frac{\omega_{yi}(I_{yy}-I_{zz})+h_y}{I_{xx}} \\ \frac{\omega_{zi}(I_{zz}-I_{xx})+h_z}{I_{yy}} & 0 & \frac{\omega_{xi}(I_{zz}-I_{xx})-h_x}{I_{yy}} \\ \frac{\omega_{yi}(I_{xx}-I_{yy})-h_y}{I_{zz}} & \frac{\omega_{xi}(I_{xx}-I_{yy})+h_x}{I_{zz}} & 0 \end{bmatrix} . \tag{B.6}$$

$\frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \mathbf{q}}$ is however much more difficult to determine. The first step is to determine which components of Equation B.2 are dependent on the attitude quaternion of the satellite. The control torque $\mathbf{N}_c$ is the sum of the torques generated by the ADCS actuators, which means that $\mathbf{N}_c$ is independent of $\mathbf{q}$. $\mathbf{N}_{gyro}$ is calculated using only the moment of inertia matrix, the angular rates and the stored angular momentum, which means that $\mathbf{N}_{gyro}$ is also independent of $\mathbf{q}$. Although there are many sources of disturbance torques, $\mathbf{N}_d$ at a LEO orbit is simplified to contain only two major components, namely gravity gradient torque ($\mathbf{N}_{gg}$) and aerodynamic torque ($\mathbf{N}_{aero}$). Even though both these components are dependent on the attitude of the satellite, only $\mathbf{N}_{gg}$ can be calculated accurately, thus

$$\mathbf{N}_d \approx \mathbf{N}_{gg} . \tag{B.7}$$

$\mathbf{N}_d$ can thus easily be expressed in terms of quaternions using Equation 2.3.3 as

$$\mathrm{N}_{dx} \approx k_{gx}(2[q_2q_3 + q_1q_4])(-q_1^2 - q_2^2 + q_3^2 + q_4^2) \tag{B.8}$$

$$\mathrm{N}_{dy} \approx k_{gy}(2[q_1q_3 - q_2q_4])(-q_1^2 - q_2^2 + q_3^2 + q_4^2) \tag{B.9}$$

$$\mathrm{N}_{dz} \approx k_{gz}(2[q_1q_3 - q_2q_4])(2[q_2q_3 + q_1q_4]) . \tag{B.10}$$

$\frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \mathbf{q}}$ can now be calculated as

$$\frac{\partial \dot{\boldsymbol{\omega}}_B^I}{\partial \mathbf{q}} = \mathbf{J}^{-1}\left[\frac{\partial \mathbf{N}_d}{\partial \mathbf{q}}\right] = \mathbf{K}\begin{bmatrix} \mathbf{d}_1 & \mathbf{d}_2 & \mathbf{d}_3 & \mathbf{d}_4 \end{bmatrix} , \tag{B.11}$$

where

$$\mathbf{K} = \begin{bmatrix} 2k_{gx} & 0 & 0 \\ 0 & 2k_{gy} & 0 \\ 0 & 0 & 2k_{gz} \end{bmatrix}$$

and

$$
\mathbf{d}_1 = \begin{bmatrix} \frac{-q_1 A_{23} + q_4 A_{33}}{I_{xx}} \\[2mm] \frac{-q_1 A_{13} + q_3 A_{33}}{I_{yy}} \\[2mm] \frac{q_3 A_{23} + q_4 A_{13}}{I_{zz}} \end{bmatrix} \qquad \mathbf{d}_2 = \begin{bmatrix} \frac{-q_2 A_{23} + q_3 A_{33}}{I_{xx}} \\[2mm] \frac{-q_2 A_{13} - q_4 A_{33}}{I_{yy}} \\[2mm] \frac{-q_4 A_{23} + q_3 A_{13}}{I_{zz}} \end{bmatrix}
$$

$$
\mathbf{d}_3 = \begin{bmatrix} \frac{q_3 A_{23} + q_2 A_{33}}{I_{xx}} \\[2mm] \frac{q_3 A_{13} + q_1 A_{33}}{I_{yy}} \\[2mm] \frac{q_1 A_{23} + q_2 A_{13}}{I_{zz}} \end{bmatrix} \qquad \mathbf{d}_4 = \begin{bmatrix} \frac{q_4 A_{23} + q_1 A_{33}}{I_{xx}} \\[2mm] \frac{q_4 A_{13} - q_2 A_{33}}{I_{yy}} \\[2mm] \frac{-q_2 A_{23} + q_1 A_{13}}{I_{zz}} \end{bmatrix} .
$$

$\frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}_B^I}$ and $\frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}}$ can be determined by partially deriving the time derivative of $\mathbf{q}$, which is

$$
\dot{\mathbf{q}} = \frac{1}{2} \underbrace{\begin{bmatrix} 0 & \omega_{zo} & -\omega_{yo} & \omega_{xo} \\ -\omega_{zo} & 0 & \omega_{xo} & \omega_{yo} \\ \omega_{yo} & -\omega_{xo} & 0 & \omega_{zo} \\ -\omega_{xo} & -\omega_{yo} & -\omega_{zo} & 0 \end{bmatrix}}_{\boldsymbol{\Omega}(\boldsymbol{\omega}_B^O)} \mathbf{q} . \tag{B.12}
$$

The relationship between $\boldsymbol{\omega}_B^I$ and $\boldsymbol{\omega}_B^O$ is given by

$$
\boldsymbol{\omega}_B^O = \boldsymbol{\omega}_B^I - \mathbf{A} \begin{bmatrix} 0 \\ -\omega_o \\ 0 \end{bmatrix} = \begin{bmatrix} \omega_{xi} + \omega_o A_{12} \\ \omega_{yi} + \omega_o A_{22} \\ \omega_{zi} + \omega_o A_{32} \end{bmatrix} , \tag{B.13}
$$

which means that $\frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}_B^I}$ can be determined as

$$
\frac{\partial \dot{\mathbf{q}}}{\partial \boldsymbol{\omega}_B^I} = \begin{bmatrix} \frac{\partial \dot{q}_1}{\partial \omega_{xi}} & \frac{\partial \dot{q}_1}{\partial \omega_{yi}} & \frac{\partial \dot{q}_1}{\partial \omega_{zi}} \\[2mm] \frac{\partial \dot{q}_2}{\partial \omega_{xi}} & \frac{\partial \dot{q}_2}{\partial \omega_{yi}} & \frac{\partial \dot{q}_2}{\partial \omega_{zi}} \\[2mm] \frac{\partial \dot{q}_3}{\partial \omega_{xi}} & \frac{\partial \dot{q}_3}{\partial \omega_{yi}} & \frac{\partial \dot{q}_3}{\partial \omega_{zi}} \\[2mm] \frac{\partial \dot{q}_4}{\partial \omega_{xi}} & \frac{\partial \dot{q}_4}{\partial \omega_{yi}} & \frac{\partial \dot{q}_4}{\partial \omega_{zi}} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \hat{q}_4 & -\hat{q}_3 & \hat{q}_2 \\ \hat{q}_3 & \hat{q}_4 & -\hat{q}_1 \\ -\hat{q}_2 & \hat{q}_1 & \hat{q}_4 \\ -\hat{q}_1 & -\hat{q}_2 & -\hat{q}_3 \end{bmatrix} . \tag{B.14}
$$

$\frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}}$ can be determined by substituting Equation B.13 into Equation B.12, which delivers

$$
\dot{q}_1 = \frac{1}{2} \Big(q_2(\omega_{zi} - \omega_o A_{32}) - q_3(\omega_{yi} - \omega_o A_{22}) + q_4(\omega_{xi} - \omega_o A_{12}) \Big) \tag{B.15}
$$

$$
\dot{q}_2 = \frac{1}{2} \Big(-q_1(\omega_{zi} - \omega_o A_{32}) + q_3(\omega_{xi} - \omega_o A_{12}) + q_4(\omega_{yi} - \omega_o A_{22}) \Big) \tag{B.16}
$$

$$
\dot{q}_3 = \frac{1}{2} \Big(q_1(\omega_{yi} - \omega_o A_{22}) - q_2(\omega_{xi} - \omega_o A_{12}) + q_4(\omega_{zi} - \omega_o A_{32}) \Big) \tag{B.17}
$$

$$
\dot{q}_4 = \frac{1}{2} \Big(-q_1(\omega_{xi} - \omega_o A_{12}) - q_2(\omega_{yi} - \omega_o A_{22}) - q_3(\omega_{zi} - \omega_o A_{32}) \Big) . \tag{B.18}
$$

By partially deriving Equations B.15 to B.18 and performing some mathematical manipulation, $\frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}}$ can be calculated as

$$
\frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}} \;=\; \frac{1}{2}\Big[\Omega(\boldsymbol{\omega}_B^O)\Big] + \omega_o
\begin{bmatrix}
\hat{q}_1\hat{q}_3 & \hat{q}_1\hat{q}_4 & 1-\hat{q}_1^2 & -\hat{q}_1\hat{q}_2 \\[4pt]
\hat{q}_2\hat{q}_3 & \hat{q}_2\hat{q}_4 & -\hat{q}_1\hat{q}_2 & 1-\hat{q}_2^2 \\[4pt]
-(1-\hat{q}_3^2) & \hat{q}_3\hat{q}_4 & -\hat{q}_1\hat{q}_3 & -\hat{q}_2\hat{q}_3 \\[4pt]
\hat{q}_3\hat{q}_4 & -(1-\hat{q}_4^2) & -\hat{q}_1\hat{q}_4 & -\hat{q}_2\hat{q}_4
\end{bmatrix}. \tag{B.19}
$$

Measurement Perturbation Jacobian Matrix

This section will show the derivation of the Jacobian matrix $\mathbf{H}_k$ that must be constructed in the execution of the full state EKF, as described in Section 5.4.

The discrete measurement perturbation matrix $\mathbf{H}_k$ from Equation 5.4.12 can be determined by partially deriving the non-linear function $\boldsymbol{h}(\mathbf{x}_k)$, which is given by Equation 5.4.15 as

$$\boldsymbol{h}(\mathbf{x}_k) \;=\; \mathbf{A}\mathbf{v}_{model,k} \;. \tag{B.20}$$

Since $\mathbf{A}$ is constructed from $\mathbf{q}$ only, Equation B.20 suggests that $\boldsymbol{h}(\mathbf{x}_k)$ is independent of $\boldsymbol{\omega}_B^I$, thus

$$\mathbf{H}_k \;=\; \begin{bmatrix} 0 & 0 & 0 & \frac{\partial h_1}{\partial q_1} & \frac{\partial h_1}{\partial q_2} & \frac{\partial h_1}{\partial q_3} & \frac{\partial h_1}{\partial q_4} \\[2mm] 0 & 0 & 0 & \frac{\partial h_2}{\partial q_1} & \frac{\partial h_2}{\partial q_2} & \frac{\partial h_2}{\partial q_3} & \frac{\partial h_2}{\partial q_4} \\[2mm] 0 & 0 & 0 & \frac{\partial h_3}{\partial q_1} & \frac{\partial h_3}{\partial q_2} & \frac{\partial h_3}{\partial q_3} & \frac{\partial h_3}{\partial q_4} \end{bmatrix}_{\mathbf{q}=\hat{\mathbf{q}}} \;. \tag{B.21}$$

$\mathbf{H}_k$ can thus be calculated as

$$\mathbf{H}_k \;=\; \begin{bmatrix} \mathbf{0}_{[3\times3]} & \mathbf{h}_1 & \mathbf{h}_2 & \mathbf{h}_3 & \mathbf{h}_4 \end{bmatrix} \;, \tag{B.22}$$

$$\text{where} \quad \mathbf{h}_1 \;=\; 2 \begin{bmatrix} q_1 & q_2 & q_3 \\ q_2 & -q_1 & q_4 \\ q_3 & -q_4 & -q_1 \end{bmatrix} \mathbf{v}_{model,k} \;,$$

$$\mathbf{h}_2 \;=\; 2 \begin{bmatrix} -q_2 & q_1 & -q_4 \\ q_1 & q_2 & q_3 \\ q_4 & q_3 & -q_2 \end{bmatrix} \mathbf{v}_{model,k} \;,$$

$$\mathbf{h}_3 \;=\; 2 \begin{bmatrix} -q_3 & q_4 & q_1 \\ -q_4 & -q_3 & q_2 \\ q_1 & q_2 & q_3 \end{bmatrix} \mathbf{v}_{model,k} \;,$$

$$\text{and} \quad \mathbf{h}_4 \;=\; 2 \begin{bmatrix} q_4 & q_3 & -q_2 \\ -q_3 & q_4 & q_1 \\ q_2 & -q_1 & q_4 \end{bmatrix} \mathbf{v}_{model,k} \;.$$

System Noise Covariance Matrix

This section will show the derivation of the covariance matrix $\mathbf{Q}_k$ that must be constructed in the execution of the full state EKF, as described in Section 5.4.

The system noise covariance matrix $\mathbf{Q}_k$ can easily be determined from the continuous domain system noise covariance matrix $\mathbf{Q}_t$ if the following assumptions are made:

- The angular rate noise (due to unmodelled disturbance torques and modelling errors) is uncorrelated.

- The system noise is small enough to allow the state matrix $\mathbf{\Phi}_k$ to be approximated using only two terms without significant inaccuracies, thus $\mathbf{\Phi}_k \approx \mathbf{I} + \left[T_s\mathbf{F}_t\right]_{t=kT_s}$.

- The angular rate noise for each axis is equal, thus $\sigma_{\omega x} = \sigma_{\omega y} = \sigma_{\omega z} = \sigma_\omega$.

Given the above-mentioned assumptions, the angular rate noise covariance matrix $\mathbf{Q}_{\omega,t}$ is given as

$$\mathbf{Q}_{\omega,t} = \begin{bmatrix} \sigma_\omega^2 & 0 & 0 \\ 0 & \sigma_\omega^2 & 0 \\ 0 & 0 & \sigma_\omega^2 \end{bmatrix} . \tag{B.23}$$

$\mathbf{q}$ is completely described by the equations in Section 5.4, which means that the noise covariance of the last four states of the system ($\mathbf{Q}_{q,t}$) is simply a zero matrix, or

$$\mathbf{Q}_{q,t} = \mathbf{0}_{[4\times4]} . \tag{B.24}$$

$\mathbf{Q}_t$ can be formed from Equations B.23 and B.24 as

$$\mathbf{Q}_t = \begin{bmatrix} \mathbf{Q}_{\omega,t} & \mathbf{0}_{[3\times4]} \\ \mathbf{0}_{[4\times3]} & \mathbf{Q}_{q,t} \end{bmatrix} \tag{B.25}$$

$$= \begin{bmatrix} \mathbf{Q}_{\omega,t} & \mathbf{0}_{[3\times4]} \\ \mathbf{0}_{[4\times3]} & \mathbf{0}_{[4\times4]} \end{bmatrix} . \tag{B.26}$$

$\mathbf{F}_t$ can also be express in the form of Equation B.26 as

$$\mathbf{F}_t = \begin{bmatrix} \mathbf{F}_{11 \ [3\times3]} & \mathbf{F}_{12 \ [3\times4]} \\ \mathbf{F}_{21 \ [4\times3]} & \mathbf{F}_{22 \ [4\times4]} \end{bmatrix} . \tag{B.27}$$

$\mathbf{Q}_k$ can now be determined by converting $\mathbf{Q}_t$ to the discrete domain. Through a process of integration [21], $\mathbf{Q}_k$ is determined to be

$$\mathbf{Q}_k = T_s\mathbf{S}_1 + \frac{1}{2}T_s{}^2\mathbf{S}_2 + \frac{1}{3}T_s{}^3\mathbf{S}_3 \,, \tag{B.28}$$

$$\text{where} \quad \mathbf{S}_1 = \mathbf{Q}_t \,, \tag{B.29}$$

$$\mathbf{S}_2 = \begin{bmatrix} \mathbf{Q}_{\omega,t}\mathbf{F}_{11}^T + \mathbf{F}_{11}\mathbf{Q}_{\omega,t} & \mathbf{Q}_{\omega,t}\mathbf{F}_{21}^T \\[2mm] \mathbf{F}_{21}\mathbf{Q}_{\omega,t} & \mathbf{0}_{[4\times 4]} \end{bmatrix}_{t=kT_s} , \tag{B.30}$$

$$\text{and} \quad \mathbf{S}_3 = \begin{bmatrix} \mathbf{F}_{11}\mathbf{Q}_{\omega,t}\mathbf{F}_{11}^T & \mathbf{F}_{11}\mathbf{Q}_{\omega,t}\mathbf{F}_{21}^T \\[2mm] \mathbf{F}_{21}\mathbf{Q}_{\omega,t}\mathbf{F}_{11}^T & \mathbf{F}_{21}\mathbf{Q}_{\omega,t}\mathbf{F}_{21}^T \end{bmatrix}_{t=kT_s} . \tag{B.31}$$

The computational load of calculating $\mathbf{Q}_k$ can be reduced if the assumption is made that $\mathbf{F}_{11} << \mathbf{F}_{21}$ [21]. $\mathbf{Q}_k$ can then be simplified to

$$\mathbf{Q}_k = \begin{bmatrix} T_s\mathbf{Q}_{\omega,t} & \frac{1}{2}T_s{}^2\mathbf{Q}_{\omega,t}\mathbf{F}_{21}^T \\[2mm] \frac{1}{2}T_s{}^2\mathbf{F}_{21}\mathbf{Q}_{\omega,t} & \frac{1}{3}T_s{}^3\mathbf{F}_{21}\mathbf{Q}_{\omega,t}\mathbf{F}_{21}^T \end{bmatrix}_{t=kT_s} . \tag{B.32}$$

Measurement Noise Covariance Matrix

This section will show the derivation of the covariance matrix $\mathbf{R}_k$ that must be constructed in the execution of the full state EKF, as described in Section 5.4.

The relationship between the true measured vector $\overline{\mathbf{v}}_{meas,k}$ and the true modelled vector $\overline{\mathbf{v}}_{model,k}$ is given by Equation 5.4.13 as

$$\overline{\mathbf{v}}_{meas,k} = \mathbf{A}(\mathbf{q}_k)\overline{\mathbf{v}}_{model,k} \ .$$

It should be noted that $\mathbf{A}_k = \mathbf{A}(\mathbf{q}_k)$. The added notation, which merely implies that $\mathbf{A}$ is a function of $\mathbf{q}_k$, will prove to be useful in the remainder of this section.

The measured and modelled vectors are furthermore related to their respective true vectors through

$$\mathbf{v}_{meas,k} = \overline{\mathbf{v}}_{meas,k} + \mathbf{m}_{meas,k} \tag{B.33}$$

$$\text{and} \quad \mathbf{v}_{model,k} = \overline{\mathbf{v}}_{model,k} + \mathbf{m}_{model,k} \ . \tag{B.34}$$

If $\Delta\mathbf{q}_k$ is defined as the difference between the true quaternion $\mathbf{q}_k$ and the estimated quaternion $\hat{\mathbf{q}}_k$, or

$$\Delta\mathbf{q}_k = \mathbf{q}_k - \hat{\mathbf{q}}_k \ , \tag{B.35}$$

then Equation 5.4.13 can also be expressed as

$$\mathbf{v}_{meas,k} - \mathbf{m}_{meas,k} = \mathbf{A}(\hat{\mathbf{q}}_k + \Delta\mathbf{q}_k)\left(\mathbf{v}_{model,k} - \mathbf{m}_{model,k}\right) \ . \tag{B.36}$$

The Taylor series expansion from Equations 5.4.4 and 5.4.11 can also be used to approximate $\mathbf{A}(\hat{\mathbf{q}}_k + \Delta\mathbf{q}_k)$ as

$$\mathbf{A}(\hat{\mathbf{q}}_k + \Delta\mathbf{q}_k) \approx \mathbf{A}(\hat{\mathbf{q}}_k) + \mathbf{C}_k\Delta\mathbf{q}_k \ , \tag{B.37}$$

$$\text{where} \quad \mathbf{C}_k = \left[\frac{\partial\mathbf{A}(\hat{\mathbf{q}}_k)}{\partial\hat{\mathbf{q}}_k}\right] \ . \tag{B.38}$$

Substituting Equation B.37 into Equation B.36 delivers

$$\mathbf{v}_{meas,k} - \mathbf{m}_{meas,k} = \left(\mathbf{A}(\hat{\mathbf{q}}_k) + \mathbf{C}_k\Delta\mathbf{q}_k\right)\left(\mathbf{v}_{model,k} - \mathbf{m}_{model,k}\right) \ . \tag{B.39}$$

Given that the innovation $\mathbf{e}_k$ is defined by Equation 5.4.17 as

$$\mathbf{e}_k = \mathbf{v}_{meas,k} - \mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{v}_{model,k} \ ,$$

substitution can be used to manipulate Equation B.39 into

$$\mathbf{e}_k = (\mathbf{C}_k\mathbf{v}_{model,k} - \mathbf{C}_k\mathbf{m}_{model,k})\Delta\mathbf{q}_k + \mathbf{m}_{meas,k} - \mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k} \ . \tag{B.40}$$

If it is assumed that $\mathbf{m}_{model,k}$ and $\Delta\mathbf{q}_k$ are extremely small compared respectively to $\mathbf{v}_{model,k}$ and $\mathbf{q}_k$, then

$$\mathbf{m}_{model,k}\Delta\mathbf{q}_k \approx 0 \ .$$

Equation B.40 can thus be simplified to

$$\mathbf{e}_k = \begin{bmatrix} \mathbf{0}_{[3\times3]} & \mathbf{C}_k\mathbf{v}_{model,k} \end{bmatrix} \Delta\mathbf{x}_k + \mathbf{m}_k \ , \tag{B.41}$$

$$\text{where} \quad \mathbf{m}_k = \mathbf{m}_{meas,k} - \mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k} \ . \tag{B.42}$$

The covariance matrix $\mathbf{R}_k$ of $\mathbf{m}_k$ is defined as

$$\mathbf{R}_k = \mathbb{E}\left\{(\mathbf{m}_k)(\mathbf{m}_k)^T\right\} \tag{B.43}$$

$$= \mathbb{E}\left\{(\mathbf{m}_{meas,k} - \mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k})(\mathbf{m}_{meas,k} - \mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k})^T\right\}$$

$$= \mathbb{E}\left\{\mathbf{m}_{meas,k}\mathbf{m}_{meas,k}{}^T - \right.$$
$$\mathbf{m}_{meas,k}\mathbf{m}_{model,k}{}^T\mathbf{A}(\hat{\mathbf{q}}_k)^T -$$
$$\mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k}\mathbf{m}_{meas,k}{}^T +$$
$$\left. \mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k}\mathbf{m}_{model,k}{}^T\mathbf{A}(\hat{\mathbf{q}}_k)^T\right\}$$

$$= \mathbb{E}\left\{\mathbf{m}_{meas,k}\mathbf{m}_{meas,k}{}^T\right\} -$$
$$\mathbb{E}\left\{\mathbf{m}_{meas,k}\mathbf{m}_{model,k}{}^T\mathbf{A}(\hat{\mathbf{q}}_k)^T\right\} -$$
$$\mathbb{E}\left\{\mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k}\mathbf{m}_{meas,k}{}^T\right\} +$$
$$\mathbb{E}\left\{\mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k}\mathbf{m}_{model,k}{}^T\mathbf{A}(\hat{\mathbf{q}}_k)^T\right\} \tag{B.44}$$

where $\mathbb{E}$ indicates the expected value operator. The last term of Equation B.44 can be simplified to

$$\mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{m}_{model,k}\mathbf{m}_{model,k}{}^T\mathbf{A}(\hat{\mathbf{q}}_k)^T = \mathbf{m}_{model,k}\mathbf{m}_{model,k}{}^T \ ,$$

since $\mathbf{m}_{model,k}\mathbf{m}_{model,k}{}^T$ is a scalar value and $\mathbf{A}(\hat{\mathbf{q}}_k)\mathbf{A}(\hat{\mathbf{q}}_k)^T = 1$. If it is furthermore assumed that the measurement noise and the model noise are uncorrelated and that each noise vector has equal variance in its 3 axes, then Equation B.44 becomes

$$\mathbf{R}_k = \mathbb{E}\left\{\mathbf{m}_{meas,k}\mathbf{m}_{meas,k}{}^T\right\} +$$
$$\mathbb{E}\left\{\mathbf{m}_{model,k}\mathbf{m}_{model,k}{}^T\right\}$$

$$= \left(\sigma_{meas}^2 + \sigma_{model}^2\right)\mathbf{I}_{3\times3} \ , \tag{B.45}$$

where σ_{meas} and σ_{model} are the respective standard deviations of $\mathbf{m}_{meas,k}$ and $\mathbf{m}_{model,k}$. It is also assumed that σ_{meas} and σ_{model} are constant.

Appendix C

Cross-Product Magnetic Controller Derivation

This appendix will provide the derivation of the Cross-Product magnetic controller from Section 6.2.3. The required magnetic moment $\mathbf{M}$ must be determined from Equation 6.2.9, or

$$-\mathbf{e} \;=\; \mathbf{M} \times \mathbf{B}_b \;.$$

The mathematical manipulation required to determine $\mathbf{M}$ can be assisted by two known mathematical relationships:

(1) If $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ are three arbitrary vectors, then

$$\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) \;=\; (\mathbf{a} \bullet \mathbf{c})\mathbf{b} - (\mathbf{a} \bullet \mathbf{b})\mathbf{c} \;. \tag{C.1}$$

(2) If $\mathbf{a}$ and $\mathbf{b}$ are two arbitrary vectors, then

$$\mathbf{a} \times \mathbf{b} \;=\; -(\mathbf{b} \times \mathbf{a}) \;. \tag{C.2}$$

If the cross products between $\mathbf{B}_b$ and both sides of Equation 6.2.9 are taken, then Equation 6.2.9 becomes

$$\mathbf{B}_b \times (-\mathbf{e}) \;=\; \mathbf{B}_b \times (\mathbf{M} \times \mathbf{B}_b) \;,$$

which can be rewritten using Equations C.1 and C.2 as

$$\mathbf{e} \times \mathbf{B}_b \;=\; (\mathbf{B}_b \bullet \mathbf{B}_b)\mathbf{M} - (\mathbf{B}_b \bullet \mathbf{M})\mathbf{B}_b \;. \tag{C.3}$$

The generated magnetic torque will be maximised if it is furthermore chosen that $\mathbf{M} \perp \mathbf{B}_b$, which will also mean that $\mathbf{B}_b \bullet \mathbf{M} = 0$. Equation C.3 can thus be simplified to

$$\mathbf{e} \times \mathbf{B}_b \;=\; \|\mathbf{B}_b\|^2 \mathbf{M} \;. \tag{C.4}$$

The required magnetic moment can now be calculated by rewriting Equation C.4 as

$$\mathbf{M} \quad = \quad \frac{\mathbf{e} \times \mathbf{B}_b}{\|\mathbf{B}_b\|^2} \; .$$