FAULT TOLERANT MULTI-ALGORITHMIC ATTITUDE DETERMINATION AND CONTROL SYSTEM FOR SMALL SATELLITES BASED ON BAYESIAN NETWORKS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

CANSU YILDIRIM

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
AEROSPACE ENGINEERING

AUGUST 2023

Approval of the thesis:

FAULT TOLERANT MULTI-ALGORITHMIC ATTITUDE DETERMINATION AND CONTROL SYSTEM FOR SMALL SATELLITES BASED ON BAYESIAN NETWORKS

submitted by **CANSU YILDIRIM** in partial fulfillment of the requirements for the degree of **Master of Science in Aerospace Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Serkan Özgen
Head of Department, **Aerospace Engineering** _____

Assoc. Prof. Dr. Halil Ersin Söken
Supervisor, **Aerospace Engineering** _____

Examining Committee Members:

Assist. Prof. Dr. Ali Türker Kutay
Aerospace Engineering, METU _____

Assoc. Prof. Dr. Halil Ersin Söken
Aerospace Engineering, METU _____

Prof. Dr. Umut Orguner
Electrical and Electronics Engineering, METU _____

Prof. Dr. İlkay Ulusoy
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Demet Çilden Güler
Astronautical Engineering, ITU _____

Date:22.08.2023

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Cansu Yıldırım

Signature :

ABSTRACT

FAULT TOLERANT MULTI-ALGORITHMIC ATTITUDE DETERMINATION AND CONTROL SYSTEM FOR SMALL SATELLITES BASED ON BAYESIAN NETWORKS

Yıldırım, Cansu
M.S., Department of Aerospace Engineering
Supervisor: Assoc. Prof. Dr. Halil Ersin Söken

August 2023, 187 pages

The Attitude Determination and Control System (ADCS) in satellites uses different modes to estimate and control attitude during various mission phases. This subsystem is important subsystem for the small satellites allowing for addressing constraints specific to small spacecraft. This thesis presents an ADCS design for an Earth-observing small satellite, including different modes with controllers and estimators. Mode transitions are achieved using threshold values and a discrete state Markov Chain model, with Bellman optimization controlling the Markov chain. Additionally, a Fault Detection and Isolation (FDI) algorithm is proposed, utilizing a Bayesian Network (BN) constructed with expert knowledge. A simulation environment is created to evaluate the performance of the algorithms and modes.

Keywords: Attitude Determination and Control System, Bayesian Network, Discrete State Markov Chain

ÖZ

KÜÇÜK UYDULAR İÇİN BAYES AĞLARINA DAYALI HATA TOLERANSLI ÇOKLU ALGORİTMALI YÖNELİM BELİRLEME VE KONTROL SİSTEMİ

Yıldırım, Cansu
Yüksek Lisans, Havacılık ve Uzay Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Halil Ersin Söken

Ağustos 2023 , 187 sayfa

Küçük uyduların Yönelim Belirleme ve Kontrol Sisteminde (YBKS) görevin farklı aşamalarında yönelimi kestirmek ve kontrol etmek için birden fazla mod kullanılmaktadır. YBKS küçük uydulara özel kısıtlamalardan etkilendiğinden dolayı önemli bir alt sistemdir. Bu tezde birden fazla mod, yönelim kestirim ve kontrol algoritması kullanılarak yer gözlem uyduları için YBKS tasarımı yapılmıştır. Modlar arası geçişler eşik değerleri ve Markov zinciri kullanılarak gerçekleştirilmiştir. Markov zinciri Bellman optimizasyonu kullanılarak kontrol edilmiştir. Ek olarak, arıza/hata tespiti ve ayıklanması (HTA) Bayes ağı yardımıyla yapılmıştır. Bayes ağının kurulması ve olasılıkların belirlenmesinde uzman bilgisinden yararlanılmıştır. Algoritmaların ve modların performanslarının anlaşılması amacıyla simülasyon kurulmuştur.

Anahtar Kelimeler: Yönelim Belirleme ve Kontrol Sistemi, Bayes Ağı, Markov Zinciri

To my persevering sister, my cheerful mother, and my tenacious father.

ACKNOWLEDGMENTS

I am grateful to my supervisor, Assoc. Prof. Dr. Halil Ersin Söken, for guiding me through the thesis preparation. His guidance and feedback have played a crucial role in improving my research.

I am acknowledging Roketsan for the academic opportunity and pleased to thank my colleagues: Ebu Bekir Çiftçi, Elif Koç, Neşe Başak Bingöl Taştankaya, and Nur Sıla Eroğlu.

I am also thankful to Berna Kaya, Cenk Mavruk, Eda Uzun Uysal, Ezgi Üstün, Gökçe Göksel, Muhammed Mustafa Balcılar, Süleyman Geçer, Uğur Sedef and İlayda Şimşek for their friendship during the this period.

I am grateful to my friends Muhammed Topalca, Muhammed Barışcan Akçin and Ceren Alkan Akçin for remaining strong despite experiencing a major earthquake in Gaziantep and Hatay. Their resilience will serve as an inspiration to me throughout my life.

I want to express my appreciation to Derya Sarmısak, Olcay Nurtaç Deniz, and Semra Sultan Uzun for never making me feel alone and always being guides in my life. I would like to extend special thanks to Semra for dedicating her time to discuss technical matters with me and for sharing my enthusiasm for engineering.

I want to express my gratitude to my sister Sude Yıldırım for her determination in facing numerous challenges at a young age. I truly believe she will accomplish all her aspirations. I want to convey my thanks to my mother for raising me with love and teaching me the value of rational thinking. Additionally, I appreciate her for teaching me the ability to find humor even during my most challenging moments. I want to thank my father for teaching me never to give up in life and for showing me that life is not just about black and white but also about various colors. I will never forget the effort he put into my education, and I will always be thankful and remember.

I want to convey my thanks to my love, Cuma Umur Doğan. I appreciate him for loving me as I am and for being a light even in my darkest moments. I am forever thankful for being my closest friend.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

LIST OF ABBREVIATIONS

ADCS	Attitude Determination and Control System
FDI	Fault Detection Isolation
MEKF	Multiplicative Extended Kalman Filter
QUEST	Quaternion Estimator Algorithm
TPM	Transition Probability Matrix
BN	Bayesian Network
PD	Proportional Derivative
EKF	Extended Kalman Filter
LQR	Linear Quadratic Regulator
UKF	Unscented Kalman Filter
PID	Proportional Integral Derivative
PI	Proportional Integral
LQG	Linear Quadratic Gaussian
TRIAD	TRIaxial Attitude Determination
MEMS	Micro-Electromechanical Systems
FOG	Fiber-Optic Gyroscope
ECI	Earth Centered Inertial Frame
LVLH	Local Vertical Local Horizontal Frame
DCM	Direction Cosine Matrix
IGRF	International Geomagnetic Reference Field
PGM	Probabilistic Graphical Model
MAP	Maximum a Posteriori
DAG	Directed Acyclic Graph

LIST OF SYMBOLS

$[\boldsymbol{q}\otimes]$ The vector-to-matrix operator transforms a vector to matrix which is used in the quaternion multiplication.

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{q}_{1:3}^T & q_3 \end{bmatrix}^T \Rightarrow [\boldsymbol{q}\otimes] = \begin{bmatrix} q_4 I_3 - [\boldsymbol{q_{1:3}}\times] & \boldsymbol{q_{1:3}} \\ -\boldsymbol{q_{1:3}}^T & q_4 \end{bmatrix}$$

$[\boldsymbol{x}\times]$ The vector-to-matrix operator transforms a vector into a skew-symmetric matrix.

$$\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}^T \Rightarrow [\boldsymbol{x}\times] = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}$$

$\boldsymbol{\omega}_X^{YZ}$ Angular velocity Y with respect to Z and resolved at X, "I" or "R" for ECI, "O" for LVLH and "B" for to body frame.

$\boldsymbol{q}$ quaternion vector, last component taken as a scalar

$$\boldsymbol{q} = \begin{bmatrix} \boldsymbol{q_{1:3}} \\ q_4 \end{bmatrix}, \quad \boldsymbol{q_{1:3}} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}$$

$\boldsymbol{r}$ Position of the spacecraft with respect to ECI frame

$\boldsymbol{v}$ Velocity of the spacecraft with respect to ECI frame

$\boldsymbol{v}_X$ Velocity of the spacecraft with respect to ECI frame resolved at X, "I" or "R" for ECI, "O" for LVLH and "B" for to body frame.

$\boldsymbol{x}$ Vector or true value of the $\boldsymbol{x}$

$\boldsymbol{x}^{(n)}$ n^{th} order derivative of the $\boldsymbol{x}$

$\boldsymbol{x}_c$ or $\boldsymbol{x}_d$ Commanded or desired value of the $\boldsymbol{x}$

$\hat{\boldsymbol{x}}$	Estimated value of the $\boldsymbol{x}$
$\hat{\boldsymbol{x}}_{k\|k}$	Expectation of the $\boldsymbol{x}$ when $\boldsymbol{y}_{0:k}$ is given, $E[\boldsymbol{x}_k \mid \boldsymbol{y}_{0:k}]$
$\hat{\boldsymbol{x}}_{k+1\|k}$	Expectation of the $\boldsymbol{x}_{k+1}$ when $\boldsymbol{y}_{0:k}$ is given, $E[\boldsymbol{x}_{k+1} \mid \boldsymbol{y}_{0:k}]$
$\Sigma_{k\|k}$	Covariance of the $\boldsymbol{x}$ when $\boldsymbol{y}_{0:k}$ is given, $cov[\boldsymbol{x}_k \mid \boldsymbol{y}_{0:k}]$
$\Sigma_{k+1\|k}$	Covariance of the $\boldsymbol{x}_{k+1}$ when $\boldsymbol{y}_{0:k}$ is given, $cov[\boldsymbol{x}_{k+1} \mid \boldsymbol{y}_{0:k}]$
$\tilde{\boldsymbol{x}}$	Measured value of the $\boldsymbol{x}$ or trajectory tracking error which is difference between current and desired state in the sliding-mode controller.
A	Matrix
$A(\boldsymbol{e}_1, \theta)$	Rotation or transformation matrix representation of the rotation around $\boldsymbol{e}_1$ unit vector with angle θ.
A_{CD}	DCM transforms the resolved vector from frame D to frame C, “I” or “R” for ECI, “O” for LVLH and “B” for the body frame.
$adj(A)$	Adjoint of matrix A
$D_Y \boldsymbol{y}$	Time derivative of the vector $\boldsymbol{y}$ with respect to frame Y, “I” or “R” for ECI, “O” for LVLH and “B” for to body frame.
$det(A)$	Determinant of matrix A
$F_X\{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3\}$	Coordinate frame X with basis vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3$, “I” or “R” for ECI, “O” for LVLH and “B” for to body frame.
$tr(A)$	Trace of matrix A

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

A small spacecraft is defined as one that has mass less than 180 kg, [37]. These spacecrafts, which were originally used for educational purposes, are suitable for testing new algorithm and mission concepts, [41]. The reason behind this is that small satellites are distinguished by their small size, light weight, and limited power, making them a more cost-effective alternative to larger satellites. Small spacecraft design, on the other hand, presents unique challenges, such as a limited selection of components, particularly when commercial off-the-shelf components are used. Additionally, small satellites frequently lack redundant components, increasing the risk of mission failure and critical data loss, according to [46]. The trend of using small satellites to carry out tasks previously handled by larger satellites, like controlling where observations are made and how accurate the data is, has increased the complexity, size, and cost of platforms, [47]. For instance, precise requirements for observable data with high spatial or time resolution over a wide range may result in increased data output from the payload, affecting the mass, volume, and budget of the subsystem. As a result, limitations and trade-offs must be considered when designing and building small satellites.

The effects of using commercial-off-the-shelf components, lack of redundancy, and complex requirements can be observed in the success rate of small spacecraft missions. Modern small satellite missions fail about half the time, [31]. In the event of a malfunction or fault in one of the spacecraft's subsystems, the mission may fail totally or partially. Consequently, each subsystem is significant. Among these subsystems, the Attitude Determination and Control System (ADCS) often plays a crucial role. Ac-

cording to the results reported in [32], 2% of space mission failures between 2009 and 2019 were due to ADCS issues. Other failure factors are design, communications, programming, and unknown factors. As a result, even within this broad classification, the ADCS is evaluated under a specific subcategory. However, it is important to note that this reference does not specifically focus on small spacecraft. Furthermore, reference [31], notably Appendix A, provides additional insights into the reasons of partially and totally unsuccessful small spacecraft between 2000 and 2016. Approximately 11% of the failures are related to issues with the sensors used in the ADCS and direct failures within the ADCS system, as determined by an analysis of failure causes. This percentage was determined by the author of the this thesis after analyzing the failure reasons in Appendix A and was not provided explicitly by the source. When faced with such circumstances, the ADCS for small spacecraft should contain several estimation and controller methods. The usage of various estimators and controllers is referred to as spacecraft modes in the literature [41, 45, 53]. These modes can be seen as a combination of estimators and controllers, forming a hybrid multi-algorithmic structure. The use of several modes provides the flexibility to meet varying accuracy requirements for different mission phases while also acting as a backup in the case of algorithm, sensor, or actuator failure. The ability to detect, isolate, and recover from faults is also critical. Understanding the occurrence of faults, detecting them, determining their address, isolation, and recovering from them are significant for small spacecraft. This increases the spacecraft's life and mission time and minimizes potential data loss, [70]. When a failure occurs, the spacecraft is often placed in a safe mode in which only critical functions are operative. This stops the spacecraft from performing its primary mission. By implementing a multi-algorithmic structure consisting of distinct modes and fault detection and isolation (FDI) capabilities, the spacecraft can avoid immediately entering safe mode and continue doing its mission objectives.

Another crucial aspect for small spacecraft is the transition between modes or satellite task planning. This is especially important when the satellite has numerous task to complete, and the completion of these tasks is strongly tied to mode changes. As stated previously, these modes are customized to the particular requirements of each task. As discussed in [36, 45], the standard method for addressing this issue in the literature is to employ threshold values. However, as underlined in [20], the usage

of threshold values may not be suited for dealing with complex scheduling or task planning.

This thesis focuses on the design of the ADCS for an Earth-observing small satellite in a sun-synchronous orbit. Magnetometer, sun-sensor, gyroscope, and star tracker are selected sensors, while reaction wheels and magnetorquers are selected actuators for the mission. Several ADCS modes are designed to handle distinct mission phases. The first mode is detumbling, which aims to reduce the initial angular velocity caused by the launch system. This mode uses the QUEST (Quaternion Estimator Algorithm) for attitude estimation and the B-dot controller for attitude control. The second mode is the mid-phase mode, which is located between the detumbling and the primary mission mode. The Multiplicative Extended Kalman Filter (MEKF) is utilized for estimator in this mode, while the B-dot controller remains as the controller. Because the MEKF and B-dot controller function independently in mid-phase mode, the term "mid-phase" is appropriate. Specifically, their individual results are not used or shared with each other. The third mode is the main mission mode. For Earth-pointing, the sliding-mode controller is used with the MEKF. This mode offers accurate estimate and attitude control. The desaturation of the reaction wheels is the primary objective of the fourth mode. The momentum dumping controller is used, with MEKF. Finally, the safe mode is used as a sun-pointing mode for battery charging process and a safety mode for emergency situation. In this mode, the sliding-mode controller is used as the controller algorithm. Mode transitions are facilitated using a both of threshold values and optimal control of discrete state Markov Chains. The dynamic behavior of the transitions is modeled as a discrete state Markov Chain, with the modes themselves serving as inputs to the dynamic model. Since modes are taken as input, mode selection is done with Bellman optimization. Transition Probability Matrices (TPMs) are taken as deterministic, and cost functions are constructed with the help of threshold values. The main purpose of the mode transition, also with Bellman optimization, is to show the applicability of the concept and give insight about future more complex task planning problems. Furthermore, as a FDI algorithm, a probabilistic graphical model known as the Bayesian Network (BN) is used. The BN aids in the selection of the filter and quaternion measurement source. In fact, two different MEKF algorithms are designed: one utilizing gyroscope measurements and the other using a mathe-

matical model. The MEKF with gyroscope measurements estimates gyroscope bias, whereas the MEKF with the mathematical model directly estimates the angular velocity. In the event of a fault, the BN selects the appropriate filter (MEKF with gyroscope measurements or MEKF with the mathematical model) and quaternion measurement source (QUEST or star tracker). Also, the BN and conditional probability distribution are constructed using expert knowledge. In summary, this thesis describes the development and evaluation of an ADCS for an Earth-observing small satellite, that includes multiple modes, mode transitions, and FDI techniques to improve mission performance and autonomy.

1.2 Literature Review

Several factors must be considered while designing the ADCS for small spacecraft, including the types of modes, methods used within those modes, sensors and actuators used in earlier designs, in-orbit experiences of other spacecraft, FDI techniques, and mode transitions. Doing a literature review based on these categorizations gives a full picture of the this research and new developments in the field.

1.2.1 Attitude Modes

SNAP-1, as the first 3-axis attitude stabilized nanosatellite [69], is designed for Earth observation. The spacecraft has two primary attitude modes: attitude acquisition and 3-axis stabilization. The goal of the attitude acquisition mode is to reducing the angular velocity of the spacecraft after launch. To achieve satellite detumbling, this mode employs B-dot control and a pitch filter. The purpose of the 3-axis stabilization mode is to maintain a stable nadir-pointing attitude toward the Earth. This mode employs a Proportional Derivative (PD) controller together with an Extended Kalman Filter (EKF). The EKF estimates state variables like angular velocity and attitude, while the PD controller adjusts the satellite's attitude depending on the desired reference attitude. ALSAT-1 is an Algerian small satellite that is used to monitor disasters. The attitude modes and algorithms of the spacecraft are quite similar to those of the SNAP-1 spacecrafts cited in [62]. Another satellite, QSAT, has three attitude modes:

detumbling, boom extension, and normal-mode phase [42]. During boom extension, a passive attitude controller is used, while in normal mode, the satellite points toward Earth. B-dot, PD, Linear Quadratic Regulator (LQR), and cross-product controllers are used on the spacecraft. The cross-product method relies on gyro and magnetometer measurements and does not require knowledge of the satellite's attitude.

The spacecraft mentioned above pragaraph were launched in the early 2000s. The following paragraph discusses more recent satellite launches. The ESTCube-2 mission encompasses various attitude modes, including detumbling, ground tracking, sun-pointing, spin-up, and orbit control [48]. ESTCube-2's spin-up phase achieves a high spin rate, a design requirement. In this mode, a PD and LQR controller the attitude. Additionally, an Unscented Kalman Filter (UKF) is employed for the estimation algorithm. Nano-JASMINE is another spacecraft with distinct attitude modes, including initial attitude control, precise attitude control, unloading phase, and observation phase, as cited in [29]. The initial attitude control mode is a detumbling mode, whereas the precise attitude control and observation phases are the principal mission phases. The unloading phase involves desaturation of the reaction wheel. Also, the spacecraft uses an EKF to adjust for magnetic disturbances, with the residual magnetic moment being one of the states within the filter. Other mission, the Small Demonstration Satellite (SDS-4), was developed by JAXA for technology demonstration purposes [45]. This spacecraft also consists of various attitude modes, including the non-operation, rate detumbling, sun acquisition, sun pointing, and Earth pointing modes. In the sun-pointing mode, the spacecraft aligns itself with the Sun in order to charge its solar panels. This mode can also be used as a safety mechanism in unexpected circumstances. In addition, the spacecraft can conduct Earth observations in Sun Pointing Mode by adjusting its orientation accordingly. The spacecraft uses Proportional Integral Derivative (PID) and EKF algorithms for attitude control and estimates. A different mission is the PicSat, which examines at star systems and needs a very accurate attitude. There are different attitude modes in the mission, such as idle, safe, measurement, detumbling, and target pointing [41]. The spacecraft is in the idle mode when it is waiting for new instructions. In the target pointing mode, methods like the Proportional Integral (PI) controller and linear Kalman Filter are used to control and estimate the spacecraft's attitude.

1.2.2 Algorithms

The Radio Aurora Explorer (RAX) is a cubesat whose goal is to study the space weather [67]. In this operation, the attitude is calculated offline at the ground station using an estimation MEKF algorithm.

The fundamental goal of the UWE-2 and UWE-3 missions is to create an attitude determination and control system using low-power commercial-off-the-shelf components [9, 57]. Due to limited power, these missions use an isotropic Kalman Filter. The B-dot controller, B-field controller, and combined wheel controller are used for control algorithms. The primary function of the B-field controller is to align the satellite's z-axis with the Earth's magnetic field. Unlike standard algorithms, the combined wheel controller employs both the magnetorquer and the reaction wheel at the same time.

Novel control and estimation techniques are investigated in the UoBSat-1 mission, [8]. Linear Quadratic Gaussian (LQG), H-infinity, and sliding-mode controllers are among the control algorithms tested. In case of wheel saturation, the LQG algorithm is used, and the spacecraft's detumbling uses an angular velocity feedback controller rather than the traditional B-dot controller. An EKF is utilized for estimating, and solar panel data is fused with the EKF, which differs from the standard practice of considering solar panel data as non-measurement.

The SiriusSat mission focuses solely on using magnetometer measurements for space weather research [30]. The EKF in this mission utilizes only magnetometer measurements. The primary aim of this work is to understand the filter performance by using a single inaccurate measurement in the space environment. One of the states in the EKF is the magnetometer bias, which is then utilized in the calculation of the magnetic damping control.

In order to evaluate the effectiveness of attitude determination and control systems in space, the ALE-2 mission was launched [58]. The MEKF is used, which makes use of several sensor measurements. Star tracker-gyroscope-EKF, geomagnetic aspect sensor-gyroscope-EKF, and geomagnetic aspect sensor-fine sun sensor-gyroscope-EKF are examples of sensor combinations to used for providing the measurements

to EKF. In this mission, different algorithms are used to calibrate the sensors at in orbit.

In the experiment described in reference [13], two types of estimation algorithms are used: the TRIaxial Attitude Determination (TRIAD) algorithm, a well-known static attitude determination algorithm, and the EKF. The EKF utilizes measurements from sun sensors and magnetometers. As expected, the performance of the filter degrades when these vector measurements become close to collinear in the orbit. The paper proposes a method to address this issue. Furthermore, the Kalman Filter uses a gyroscope, and solar sensor measurements are also given.

1.2.3 Sensors and Actuators

Based on the analysis of ADCS designs in previous Earth missions [9, 21, 29, 33, 41, 42, 45, 48, 58, 62, 67, 69, 71], a bar chart is generated to illustrate the selected sensors and actuators in the given missions. Fig. 1.1 presents this information, which gives useful data about the sensor and actuator choices that are usually used on small spacecraft missions at Earth orbit. This visualization enhances our understanding of the general trends and intuitive decision-making for sensor and actuator selection.

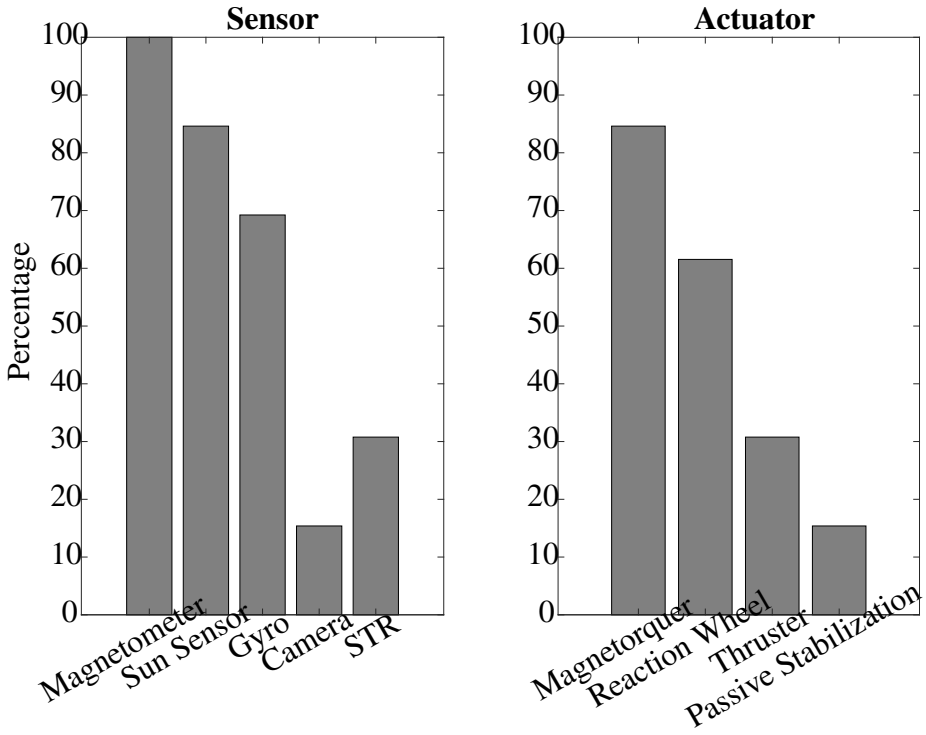


Figure 1.1: Percentage of sensors and actuators used in the small spacecraft mission, STR stands for the star tracker

1.2.4 In-orbit Problems

The ESTCube-1 satellite had a problem caused by a large magnetic residual moment, equivalent in magnitude to that produced by the reaction wheel [21]. Another issue is unsteady spin rates and orientations during in-orbit operations.

Periodic changes in the rotation axis are an issue for the UWE-3 spacecraft. The magnetometer measurements not only captured the magnetic field variations caused by orbital angular velocity but also those induced by the spacecraft's angular velocity. To address this, a magnetometer calibration algorithm was employed [15].

Two sun sensors on the Aalto-1 spacecraft failed in orbit. The frequency of gyroscope measurements was insufficient for the mission, requiring sensor algorithm updates during the operation. Additionally, problems with the spacecraft's detumbling process were observed, which were linked to the magnetorquer. A previously unknown

magnetic disturbance was also noticed, the root cause of which is unclear [43, 53].

The SNUGLITE project includes developing a LQG-based controller for satellites and testing its functionality in space. However, because of a problem with the uplink between the ground station and the spaceship, the initial standby mode could not be changed [33].

1.2.5 FDI

As described in reference [28], there are two basic techniques to FDI: hardware redundancy and analytical redundancy. Hardware redundancy involves utilizing multiple hardware components that measure the same signal to perform the FDI process. On the other hand, analytical redundancy relies on algorithms to achieve FDI. Analytical redundancy can further be divided into two categories: quantitative and qualitative, as highlighted in the article. The quantitative approach utilizes mathematical models to detect and isolate faults, while the qualitative approach uses artificial intelligence technologies. This section of the literature review will examine relevant studies based on this categorization. For instance, reference [16] presents an example of hardware redundancy in small spacecraft. Multiple sensor topologies are employed to measure a single physical signal, and the selection of sensors is determined by employing a "best" approach. Statistical methods, such as running mean and variances, are used to detect and isolate errors. In contrast, reference [61] employs analytical redundancy for spacecraft actuator fault detection. A fault observer is designed to detect and isolate actuator faults, indicating the utilization of a quantitative approach within the analytical framework. Another example of analytical redundancy can be found in references [3, 4], where a robust EKF is developed to address sensor faults and rapid spacecraft maneuvers. Fault detection is achieved through statistical methods by examining the innovation matrix of the filter and utilizing threshold values.

This study focuses on the application of BN for FDI. BN falls under the analytical redundancy qualitative approach. Reference [49] highlights that BN-based reasoning algorithms explicitly illustrate cause and effect relationships, making BN a suitable method for FDI. While the usage of BN in FDI processes for spacecraft attitude determination and control is not widely explored, this literature review examines its appli-

cation in small spacecraft, not limited to attitude determination and control systems.

In reference [54], a tool is developed to identify fault roots in satellite payloads. BN is constructed using expert knowledge and associated probability distributions. Similarly, in reference [14], BN is used to model the satellite network system. In reference [11], BN is utilized for fault and error observation in spacecraft constellations based on telemetry data. A hierarchical fault-diagnosis structure is established, encompassing sensor and actuator faults, spacecraft subsystems, individual spacecraft, and formation level. All error and fault sources are specified at each level, with interconnections established between them. A novel method called the confusion matrix is used to derive probability distributions. Reference [72] focuses on detecting momentum wheel faults using BN. The construction of the BN and acquisition of probability distributions once again relies on expert knowledge. Lastly, in the example described in [73], BN is employed to assess the system security level. Nodes in the BN include attitude determination faults and actuator faults. The construction of the BN and determination of probability distributions are based on expert knowledge.

1.2.6 Mode Transitions

In references [36, 45, 56], mode transitions are achieved using threshold values. However, more recent studies have adopted a different approach for satellite tasking operations by formulating the problem as a Markov model. In reference [20], specifically, the transition is modeled as a semi-Markov model, which removes the assumption of a uniform time step. The inputs in this model are defined as image collections, ground contacts, battery recharging, and data recorder management, representing discrete modes. Furthermore, the state space is composed of continuous variables. To address the problem, methods such as forward search and Monte Carlo Tree search are employed. In reference [26], the satellite multi-tasking process is modeled as a Markov process. The states in this model includes various parameters such as spacecraft position, velocity, L^2 norm of the attitude error and rate, reaction wheel speeds, battery charge, and other relevant factors. These states are again considered continuous variables. The inputs or actions in this model are defined as charge, desaturate, downlink, and image targeting, representing the different modes of the satellite. Bell-

man optimization and Monte Carlo Tree search are used as methods for optimization.

1.3 Contributions and Novelties

Contributions of the thesis are as follows:

- The ADCS modes and algorithms which are used in the modes are designed for a small satellite that is 700mm×700mm×500mm, with a weight of 95 kg. The main purpose of this satellite is Earth-observing.
- The process of transitioning between modes is facilitated by employing threshold values and modeling these transitions utilizing a Markov Chain. The determination of modes is achieved with the application of Bellman Optimization for the Markov Chains.
- BNs are utilized within FDI algorithms. Graphs and conditional probability distributions are constructed.

1.4 The Outline of the Thesis

This thesis is divided into two parts. The first part covers the necessary preliminaries required for understanding and developing the algorithms. The second part focuses on the design and evaluation of ADCS, FDI algorithms, and modes. The first part consists of the Chapters 2 and 3.

In Chapter 2, the influence of spacecraft mission purpose on the ADCS system is discussed. The selection process for sensors and actuators is given for the Earth-observing mission. Furthermore, this chapter includes relevant reference frames.

In Chapter 3, attitude representations and their transformations are given. Additionally, it provides satellite dynamics and kinematics equations for simulating satellite motion. The chapter also covers measurement equations and disturbance torques in space.

The second part of this thesis consists of Chapter 4, Chapter 5, Chapter 6, Chapter 7, and Chapter 8.

In Chapter 4, the controller and estimator algorithms used in this thesis are presented. The controller algorithms include the B-dot controller, sliding-mode controller, and reaction wheel desaturation controller. The estimator algorithms consist of the QUEST algorithm and the MEKF algorithm.

In Chapter 5, the proposed FDI algorithm, which is based on BN is discussed. The chapter covers the constructing process of the BN and determining the observable nodes.

In Chapter 6, a detailed explanation of the different modes in the spacecraft's ADCS is provided. It describes the specific purposes of each mode and how the controllers and estimators are used together in the modes. The chapter also discusses the methods used at transition between modes: threshold values and Bellman optimization.

In Chapter 7, the performance of each mode in the ADCS is discussed. The chapter also shows the performance of the overall ADCS, that includes the results of the FDI algorithm.

In Chapter 8, conclusion and future works are discussed.

CHAPTER 2

MISSION REQUIREMENT ANALYSIS AND SELECTION OF THE ACTUATORS AND SENSORS

Understanding the mission purpose is important in order to establish the mission requirements and develop the ADCS accordingly. The ADCS design consists of various parts, including mode design, selection of control and estimation algorithms, and choice of sensors and actuators. These are determined based on the mission's objectives and requirements. In this chapter, the mission requirements are analyzed to determine the appropriate sensors and actuators for the spacecraft.

Section 2.1 presents different spacecraft and discusses how their specific purposes influence the design of their ADCS. Then, selection of the sensors and actuators is discussed for the nadir pointing spacecraft in the Section 2.2. Also, in this part, the chosen sensors and actuators are then presented, along with brief explanations of their functions.

2.1 Mission Requirement Analysis

The pointing accuracy, as well as the selection of sensors and actuators and the design of the ADCS modes is influenced by the purpose of the mission and the limitations for the mass, size and budget. Three small satellite missions are presented in a Table 2.1. The first mission is SNAP-1 [68, 69], an Earth-observing satellite that operates in a 704 km sun-synchronous orbit. The pointing accuracy of 3 degrees at 1 sigma is sufficient for the mission. The spacecraft has two modes: Attitude acquisition mode and 3-axis stabilization mode, which are used for detumbling and nadir pointing, respectively. The spacecraft uses a magnetometer and camera for the mission, as it has relatively

simple mission requirements.

The second mission is the ESTCube-2 mission [48], which is designed to demonstrate a novel spin-up technique and serve as a precursor to the ESTCube-3 mission for the moon orbit. Given the high spin rate of 360 degrees per second, the modes must be designed accordingly, including a spin-up mode. Additionally, the mission's goal to prepare for ESTCube-3 necessitates the use of redundant sensors, including a magnetometer and magnetorquer, which have been replaced by a star tracker and cold-gas propulsion system.

The final mission is Nano-JASMINE, which aims to estimate star positions and update star catalogs in a 705 km sun-synchronous orbit [29]. High attitude rate accuracies are required for the mission, 2.3×10^{-5} deg/s, and observation mode has been designed to meet this requirement. To achieve the required accuracies, the mission utilizes two distinct types of gyroscopes: Micro-electromechanical systems (MEMS) and Fiber-Optic Gyroscope (FOG). The selection of sensors and actuators is again based on mission requirements.

Table 2.1: Impacts of the mission objective on the mission purpose, attitude pointing accuracy, ADCS modes, sensors and actuators selection, [29, 48, 68, 69]

Name	Mission Purpose	Challange / Highest Pointing Accuracy	ADCS Modes	Sensors	Actuators
SNAP-1	The main purpose of SNAP-1 is 3-axis attitude stabilization.	First, 3-axis stabilized nanosatellite /3 deg	-Attitude Acqusition - 3-axis stabilization	-Magnetometer -Camera	- Y Reaction Wheel -Magnetorquer
ESTCube-2	The ESTCube-2 mission utilized the magnetic actuators to test a new spin-up mechanism for a nanosatellite and was conducted as a preliminary step towards an interplanetary satellite in lunar orbit.	The high spin rate requirement necessitates algorithms capable of rapidly processing sensor readings and calculating outputs. / 0.25 deg and 0.125 deg/s	-Detumbling -Ground-tracking -Sun Pointing -Spin up -Orbit control	-Gyroscope -Magnetometer (for redundancy) -Sun sensor -Accelerometer -Star tracker	-Reaction Wheels -Magnetorquer (for redundancy) - Cold gas propulsion
Nano-JASMINE	The mission's objective is to update the star catalog.	Precise attitude control is required. /0.05 deg	-Initial Attitude and Coarse Control - Precise Attitude Control - Unloading - Observation Phase	-Gyroscope -Magnetometer -Sun sensor - Mission Telescope -Star tracker	- Reaction Wheel - Magnetorquer - Magnetic Canceller

The satellites used in this thesis are around 700mm×700mm×500mm in size and mass 95 kg. Also, the primary goal of the satellite in this thesis is to meet the requirements of any Earth-observation spacecraft. Therefore, there is no specific definition for attitude angle and rate accuracy. The modes and algorithms are designed to fulfill general requirements. In the following section, the selection of actuators and sensors is also carried out to serve the general purpose of nadir pointing. Furthermore, the spacecraft is conceptualized to be in a sun-synchronous orbit at an altitude of 705 km with an inclination of 98 degrees.

2.2 Selection of the Actuators and Sensors

This paragraph describes the selection process for sensors and actuators in an ADCS for a general-purpose Earth-pointing spacecraft. The requirements of the spacecraft

were taken into consideration during the selection process, and common sensors and actuators were chosen as discussed in the Section 1.2.3. Since there were no specific mission requirements for the ADCS system, a general-purpose design was used, and the sensors and actuators were chosen accordingly.

2.2.1 Sensors

This section provides an overview of the selected sensors and their explanations.

2.2.1.1 Magnetometer

The magnetometer is a widely used sensor for small spacecraft, especially in low Earth orbit. Its main purpose is to measure the Earth's magnetic field in vector form, resolved at the spacecraft body frame. This sensor is useful in the initial stages of attitude estimation, where relatively basic algorithms are employed. Moreover, when combined with other sensor measurements, the magnetometer can be used as a source of data for advanced algorithms that enable accurate attitude estimation. In fact, magnetometer providing continuous measurement without any interruption is a major advantage.

2.2.1.2 Sun Sensor

Sun sensors are primarily used for determining the direction of the sun in relation to the spacecraft's body frame, providing only a unit vector as output. In contrast, magnetometers provide both direction and magnitude of the measured magnetic field resolved at the satellite body frame. Sun sensors is a popular choice because of their ability to be used with magnetometers for initial course attitude determination algorithms. In these relatively basic algorithms, at least two vectors must be known at the body frame and inertial frame, making the combination of magnetometer and sun sensor a suitable choice. Additionally, sun sensors are useful for directing the spacecraft's solar panel towards the sun.

2.2.1.3 Gyroscope

A gyroscope is a important sensor for determining the angular velocity of a spacecraft relative to the inertial frame and resolved at the body axis. This is because the spacecraft's attitude values cannot be separated from its angular velocity. In fact any prediction for the spacecraft's attitude at time k+1 uses the attitude and angular velocity at time k. The angular velocity at time k directly affects the attitude at time k+1. Additionally, gyroscope measurements can be used during the initial phase of the spacecraft to decrease its angular velocity step-by-step using control algorithms. Furthermore, during the fine attitude determination process, gyroscope measurements can be used in combination with other sensors as a measurement source for the filter.

2.2.1.4 Star Tracker

A star tracker is a sensor that directly measures the attitude of a spacecraft by comparing the stars observed in the field of view of the camera and an already downloaded absolute catalog. It is known with its high accuracy relative to other sensors. When the spacecraft is at low angular velocities, the star tracker can be used directly to obtain attitude. Therefore, it is particularly useful during fine attitude estimation, since the spacecraft initially has high angular velocity.

2.2.2 Actuators

An overview of the chosen actuators is provided in this subsection, along with each actuators' description.

2.2.2.1 Reaction Wheels

A reaction wheel is device used for spacecraft attitude control. It operates through a spinning wheel that generates angular momentum, and a change in the angular momentum creates the torque. Compared to magnetorquers, it provides relatively high pointing accuracy at the control. However, the problem of saturation can arise when

the wheels reach their highest spin rates. To address this, the wheel spin rate must decrease, a process known as desaturation. In such cases, magnetorquers are used for the desaturation process.

2.2.2.2 Magnetorquers

Magnetorquers are a component used in spacecraft attitude control. These devices function by using electromagnetic coils to create a magnetic field, which interacts with the Earth's magnetic field to generate torque. Although magnetorquers can interfere with other magnetic field devices on the spacecraft, they are frequently used in combination with reaction wheels, especially during the desaturation process to decrease the wheels' spin rate. Despite their relatively low accuracy, magnetorquers provide essential backup and complementary capabilities to reaction wheels in spacecraft attitude control systems.

CHAPTER 3

REFERENCE FRAMES, ATTITUDE REPRESENTATIONS, SATELLITE KINEMATICS AND DYNAMICS, MEASUREMENT EQUATIONS AND EXTERNAL TORQUES

Attitude represents the orientation of an object in space with respect to a chosen reference frame. There are various attitude representations available, they require at least three attitude coordinates. Using only three coordinates, on the other hand, can result in singularities, making some attitudes unrepresentable or ambiguously represented. This can result in situations where attitudes blow up to infinity or attitudes that cannot be uniquely defined. To avoid these issues, more than three coordinates can be used, but this adds constraints to the attitude representation.

In the Section 3.1, the reference frames used in the study are introduced. Transformation matrices for frame transformations are provided, along with the corresponding angular velocities.

Section 3.2 discusses attitude representations. Section 3.3 focuses on the transformations between these representations.

After considering the advantages and disadvantages of attitude representations, quaternions are chosen in this thesis. The attitude kinematic equation for quaternions is provided in Section 3.4.

In the Section 3.5 Euler's equation of motion is given with and without the use of reaction wheels. While giving both attitude kinematics and dynamics equations, spacecraft rotation can be completely simulated.

In the Section 3.6, measurement equations for constructing measurements in the sim-

ulation environment are explained.

Finally, Section 3.7 disturbance torques in space are examined, with specific focus on the gravity-gradient.

3.1 Reference Frames

This subsection presents the frames that were used in the thesis.

3.1.1 Earth Centered Inertial Frame

Newton's laws are according to an inertial frame, which is a frame of reference that is not accelerating or rotating relative to distant stars. The Earth-Centered Inertial (ECI) Frame is commonly used as an inertial frame in space applications. Inertial sensors use the ECI frame as a reference frame. However, since the Earth's orbit around the Sun is changing, the ECI frame is not strictly inertial. Despite this, using the ECI frame is sufficient for most practical applications, as sensors are unable to detect these slight variations.

- The x-axis of the frame is oriented towards the vernal equinox, which is the one of the point of intersection between the Earth's equatorial plane and its orbit around the Sun as the ascending node.
- The z-axis of the frame corresponds to the Earth's rotational axis, which points towards the true north pole.
- The y-axis completes the right-handed triad, meaning it completes a set of three mutually perpendicular axes that form a right-handed coordinate system.

3.1.2 Local Vertical Local Horizontal Frame

Using a frame aligned with the orbit of the spacecraft is a practical approach for a nadir-pointing mission, as it can serve as the local navigation frame for the attitude

determination and control process. It should be noted that in this context, the term "orbit frame" is used interchangeably with the Local Vertical Local Horizontal (LVLH) frame.

- The z-axis of the frame is oriented towards the center of the Earth.
- The y-axis is aligned with the negative direction of the orbit normal vector.
- The x-axis completes the right-handed triad.

3.1.3 Body Frame

The body frame is a reference frame that is fixed to the spacecraft, rotating with the body and remaining fixed. The measurements from sensors such as gyroscope, magnetometer, and sun sensors are resolved in the body frame. The x, y, and z axes of the body frame can be defined according to the spacecraft's geometry and vary depending on the application, as long as they form a right-handed triad. In this thesis, the origin of the body frame is located at the center of gravity.

3.1.4 Reference Frame Transformations

In this subsection, the transformation matrices between the frames used in the thesis are presented, along with the corresponding angular velocities between frames [40]. These angular velocities are essential in obtaining the desired quaternion values for the sliding-mode control.

3.1.4.1 Local Vertical Local Horizontal Frame to Inertial Frame

The transformation matrix from the orbit frame to the inertial frame is presented in Eq. 3.1 and 3.2. Additionally, Eq. 3.3 provides the angular velocity between the two frames, resolved in the orbit frame.

$$
\begin{aligned}
\boldsymbol{o}_{3I} &= -\boldsymbol{r}_I/\|\boldsymbol{r}_I\| \\
\boldsymbol{o}_{2I} &= -(r_I \times \boldsymbol{v}_I)/\|\boldsymbol{r}_I \times \boldsymbol{v}_I\| \\
\boldsymbol{o}_{1I} &= \boldsymbol{o}_{2I} \times \boldsymbol{o}_{3I}
\end{aligned} \tag{3.1}
$$

$$
A_{IO} = \begin{bmatrix} \boldsymbol{o}_{1I} & \boldsymbol{o}_{2I} & \boldsymbol{o}_{3I} \end{bmatrix} \tag{3.2}
$$

$$
\boldsymbol{\omega}_O^{OI} = \begin{bmatrix} 0 \\ -\|\boldsymbol{r}_I \times \boldsymbol{v}_I\|/\|\boldsymbol{r}_I\|^2 \\ \|\boldsymbol{r}_I\|(\boldsymbol{o}_{2I} \cdot \dot{\boldsymbol{v}}_I)/\|\boldsymbol{r}_I \times \boldsymbol{v}_I\| \end{bmatrix} \tag{3.3}
$$

3.1.4.2 Local Vertical Local Horizontal Frame to Body Frame

The transformation matrix between the orbit frame and body frame is presented in Eq. 3.4 and 3.5 for both 321 and 313 rotations. Also, it should be emphasized that, in contrast to the classical flight mechanics terminology [22], the ψ angle and ϕ angle are interchanged in this study for 321 rotation. Specifically, the ψ angle represents the rotation around $\boldsymbol{e}_1$, while the ϕ angle represents the rotation around $\boldsymbol{e}_3$. Additionally, the angular velocity between frames is given in Eq. 3.6, resolved at the body frame. Moreover, the main aim of the ADCS is the process of these angles (ϕ, θ, ψ) determination, control, and estimation.

$$
\begin{aligned}
A_{BO} = A_{321}(\phi,\theta,\psi) &= A(\boldsymbol{e}_1,\psi)A(\boldsymbol{e}_2,\theta)A(\boldsymbol{e}_3,\phi) \\
&= \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\psi & s\psi \\ 0 & -s\psi & c\psi \end{bmatrix}\begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix}\begin{bmatrix} c\phi & s\phi & 0 \\ -s\phi & c\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} c\theta c\phi & c\theta s\phi & -s\theta \\ -c\psi s\phi + s\psi s\theta c\phi & c\psi c\phi + s\psi s\theta s\phi & s\psi c\theta \\ s\psi s\phi + c\psi s\theta c\phi & -s\psi c\phi + c\psi s\theta s\phi & c\psi c\theta \end{bmatrix}
\end{aligned} \tag{3.4}
$$

$$
\begin{aligned}
A_{BO} = A_{313}(\phi,\theta,\psi) &= A(\boldsymbol{e}_3,\psi)A(\boldsymbol{e}_1,\theta)A(\boldsymbol{e}_3,\phi) \\
&= \begin{bmatrix} c\psi & s\psi & 0 \\ -s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & c\theta & s\theta \\ 0 & -s\theta & c\theta \end{bmatrix}\begin{bmatrix} c\phi & s\phi & 0 \\ -s\phi & c\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} c\psi c\phi - s\psi c\theta s\phi & c\psi s\phi + s\psi c\theta c\phi & s\psi s\theta \\ -s\psi c\phi - c\psi c\theta s\phi & -s\psi s\phi + c\psi c\theta c\phi & c\psi s\theta \\ s\theta s\phi & -s\theta c\phi & c\theta \end{bmatrix}
\end{aligned} \tag{3.5}
$$

$$
\boldsymbol{\omega}_B^{BO} = \begin{bmatrix} \dot{\psi} - \dot{\phi}\sin\theta \\ \dot{\phi}\cos\theta\sin\psi + \dot{\theta}\cos\psi \\ \dot{\phi}\cos\theta\cos\psi - \dot{\theta}\sin\psi \end{bmatrix} \tag{3.6}
$$

3.1.4.3 Inertial Frame to Body Frame

The transformation matrix between the body frame and the inertial frame can be obtained by utilizing A_{IO} and A_{BO}, as presented in Eq. 3.7. The same applies to the angular velocity, which is given in Eq. 3.8.

$$
\begin{aligned}
A_{BI} &= A_{BO}A_{IO}^T \\
&= A_{BO}A_{OI}
\end{aligned} \tag{3.7}
$$

$$
\begin{aligned}
\boldsymbol{\omega}_B^{BI} &= \boldsymbol{\omega}_B^{BO} + A_{BO}\boldsymbol{\omega}_O^{OI} \\
&= \boldsymbol{\omega}_B^{BO} + \boldsymbol{\omega}_B^{OI}
\end{aligned} \tag{3.8}
$$

3.2 Attitude Representations

This section provides an explanation of various attitude representations.

3.2.1 Direction Cosine Matrix

The Direction Cosine Matrix (DCM) is a widely used attitude representation method that employs a transformation matrix based on the cosine angles between the unit vectors of the reference frames. In three-dimensional space, each of the two frames involved three independent unit vectors. One of the unit vectors of frame N can be expressed in terms of the three unit vectors of frame $\boldsymbol{B}$, by calculating the angles between them using dot products, which gives the cosine angles. These angles can then be used to create a transformation matrix known as the DCM. Therefore, the DCM has a total of nine attitude parameters. These relationships are illustrated in Eq. 3.9 and 3.10.

$$\begin{aligned} \boldsymbol{b}_1 &= cos\alpha_{11}\boldsymbol{n}_1 + cos\alpha_{12}\boldsymbol{n}_2 + cos\alpha_{13}\boldsymbol{n}_3 \\ \boldsymbol{b}_2 &= cos\alpha_{21}\boldsymbol{n}_1 + cos\alpha_{22}\boldsymbol{n}_2 + cos\alpha_{23}\boldsymbol{n}_3 \\ \boldsymbol{b}_3 &= cos\alpha_{31}\boldsymbol{n}_1 + cos\alpha_{32}\boldsymbol{n}_2 + cos\alpha_{33}\boldsymbol{n}_3 \end{aligned} \tag{3.9}$$

If frames N and $\boldsymbol{B}$ have canonical basis sets,

$\Rightarrow$

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} cos\alpha_{11} & cos\alpha_{12} & cos\alpha_{13} \\ cos\alpha_{21} & cos\alpha_{22} & cos\alpha_{23} \\ cos\alpha_{31} & cos\alpha_{32} & cos\alpha_{33} \end{bmatrix} \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix} \tag{3.10}$$

where

$$cos\left(\angle \boldsymbol{b}_i, \boldsymbol{n}_j\right) = \boldsymbol{b}_i \cdot \boldsymbol{n}_j$$

As mentioned before, three attitude coordinates are sufficient for attitude representations. However, the DCM has nine components, which means that it is an over-parametrized attitude representation. Therefore, the DCM must satisfy six constraints. It should be noted that the DCM represents the basis vectors of frame $F_B\{b_1, b_2, b_3\}$ in terms of frame $F_N\{n_1, n_2, n_3\}$. Since the basis vectors have a norm of one and the basis sets are canonical, meaning they are perpendicular, the following constraints can be listed:

1. The rows of the DCM must have a norm of one.
2. The columns of the DCM must have a norm of one.

3. The inner product of different rows of the DCM must be zero.

4. The inner product of different columns of the DCM must be zero.

However, these constraints are dependent on each other, and there are only six independent constraints. The properties described for the matrix, including having norm one for its rows and columns, and having orthogonal rows and columns, are collectively referred to as an orthonormal matrix. Therefore, Eq. 3.11 can be written:

$$C_{NB} = C_{BN}^{-1} = C_{BN}^{T} \tag{3.11}$$

Furthermore, since the coordinate frames are proper, which means they are right-handed, the determinant of the DCM is equal to 1.

The addition and subtraction of rotations can be easily represented using the DCM by multiplying the two DCMs, Eq. 3.12.

$$C_{RN} = C_{RB}C_{BN} \tag{3.12}$$

3.2.2 Euler Angles

Euler angles are a commonly used attitude representation that can be defined as a sequence of rotations. The order of rotations and rotation axes are crucial in Euler angles. There are 12 sets of Euler angles, which can be divided into symmetric and asymmetric sets. Both symmetric and asymmetric sets have unique and common properties. Since Euler angles have three attitude parameters, this representation suffers from singularity problems. The singularity issue occurs at the second angle for both symmetric and asymmetric sets. However, for the asymmetric set, this ambiguity occurs at $\pm\pi/2$, while for the symmetric set, it occurs at 0 and π. In this thesis, both symmetric and asymmetric sets are used in Section 3.1.4.2. In addition, to perform addition and subtraction of Euler angles, first the Euler angles must be transformed into a DCM; then, the addition or subtraction can be performed using the DCM. Finally, the resulting DCM is retransformed to Euler angles.

3.2.3 Quaternion

The quaternion is another parameter used for attitude representation, composed of a vector and a scalar part, as shown in Eq. 3.13. In this thesis, the scalar part of the quaternion is taken as the last component. The vector part of the quaternion (e_1, e_2, e_3) represents the rotation axis and it is invariant under the transformations, Eq. 3.14. This means that it has the same representation in both frames.

$$\boldsymbol{q} = \begin{bmatrix} \boldsymbol{q_{1:3}} \\ q_4 \end{bmatrix}, \quad \boldsymbol{q_{1:3}} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{3.13}$$

$$\boldsymbol{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} e_1 \sin(\theta/2) \\ e_2 \sin(\theta/2) \\ e_3 \sin(\theta/2) \\ \cos(\theta/2) \end{bmatrix} \tag{3.14}$$

Since quaternions use four attitude coordinates, there is no singularity issue with this representation. However, this means that a quaternion norm constraint is necessary, which requires the norm of the quaternion to be equal to one. Additionally, since quaternions are four-dimensional and their norms are one, they represent a hypersurface in four-dimensional space. This means that a unique attitude can be represented by two different quaternion sets, which are directly opposite of each other at a hyperplane. Therefore, the quaternion representation is not unique. Geometrically, this means that one quaternion represents a short rotation and the other represents a long rotation. The short and long rotations are important from a control perspective, as finding the desired quaternion with a long path can require more control effort.

Quaternion addition and subtraction can be performed easily without the need for intermediate transformations. However, in this thesis, to maintain consistency, the quaternion to DCM transformation is utilized prior to addition or subtraction. After the operation, the resulting DCM is then transformed back to quaternions.

Quaternion multiplication and inverse operations are utilized in the thesis. However,

instead of Hamilton's quaternion multiplication, Eq. 3.15 is used for quaternion multiplication [40]. Also, the multiplication of a quaternion with its inverse results in the unit quaternion, denoted as $\boldsymbol{I}_q = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$, as shown in Eq. 3.16.

$$\bar{\boldsymbol{q}} \otimes \boldsymbol{q} = \begin{bmatrix} q_4\bar{\boldsymbol{q}}_{1:3} + \bar{q}_4\boldsymbol{q}_{1:3} - \bar{\boldsymbol{q}}_{1:3} \times \boldsymbol{q}_{1:3} \\ \bar{q}_4 q_4 - \bar{\boldsymbol{q}}_{1:3} \cdot \boldsymbol{q}_{1:3} \end{bmatrix} \tag{3.15}$$

$$\begin{aligned} \boldsymbol{q} \otimes \boldsymbol{q}^{-1} &= \boldsymbol{I}_q \\ \boldsymbol{q}^{-1} &= \begin{bmatrix} -\boldsymbol{q}_{1:3} \\ q_4 \end{bmatrix} \Big/ \|\boldsymbol{q}\|^2 \end{aligned} \tag{3.16}$$

3.3 Transformation Between Attitude Representations

This subsection covers the transformation between different attitude representations.

3.3.1 Euler Angles to DCM

To obtain the DCM representation of a given sequence of rotations, one needs to have access to the basic rotation matrices. These matrices are crucial for constructing complex rotations, with each of them representing a rotation around a single axis. For the three-dimensional space, rotation matrices around the first, second, and third axes are commonly used, and are presented at Eq. 3.17.

$$\begin{aligned} A(\boldsymbol{e}_1, \theta) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta & -sin\theta \\ 0 & sin\theta & cos\theta \end{bmatrix} \\ A(\boldsymbol{e}_2, \theta) &= \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix} \\ A(\boldsymbol{e}_3, \theta) &= \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{3.17}$$

For instance, if a rotation sequence is given in Eq. 3.18, the DCM representation can be obtained using Eq. 3.19 .

$$F_A \xrightarrow{rot[e_3,\psi]} F_P \xrightarrow{rot[e_2,\theta]} F_Q \xrightarrow{rot[e_1,\phi]} F_B \tag{3.18}$$

$$\begin{aligned} A_{AB} &= A(e_3,\psi)A(e_2,\theta)A(e_1,\phi) = A(\psi,\theta,\phi) \\ &= \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & -sin\phi \\ 0 & sin\phi & cos\phi \end{bmatrix} \end{aligned} \tag{3.19}$$

3.3.2 DCM to Euler Angles

Extraction of Euler angles from the DCM is demonstrated for two types of Euler sequences: the symmetric 313 rotation sequence and the asymmetric 321 rotation sequence [50]. Furthermore, the derivation of this DCM is presented in Section 3.1.4.2.

$$A_{321}(\phi,\theta,\psi) = \begin{bmatrix} c\theta c\phi & c\theta s\phi & -s\theta \\ -c\psi s\phi + s\psi s\theta c\phi & c\psi c\phi + s\psi s\theta s\phi & s\psi c\theta \\ s\psi s\phi + c\psi s\theta c\phi & -s\psi c\phi + c\psi s\theta s\phi & c\psi c\theta \end{bmatrix} \tag{3.20}$$

1. $c_{13} = -sin\theta \Rightarrow cos\theta = \sigma\sqrt{1 - c_{13}^2}$ where $\sigma = \pm 1$
   $\theta = atan2(-c_{13}, \sigma\sqrt{1 - c_{13}^2})$
   $\sigma = 1$ leads to θ is acute angle and $\sigma = -1$ leads to θ is obtuse angle. Generally, $\sigma = 1$ is preferred.

2. $\psi = atan2(c_{23}, c_{33})$

3. $\phi = atan2(c_{12}, c_{11})$

$$A_{313}(\phi,\theta,\psi) = \begin{bmatrix} c\psi c\phi - s\psi c\theta s\phi & c\psi s\phi + s\psi c\theta c\phi & s\psi s\theta \\ -s\psi c\phi - c\psi c\theta s\phi & -s\psi s\phi + c\psi c\theta c\phi & c\psi s\theta \\ s\theta s\phi & -s\theta c\phi & c\theta \end{bmatrix} \tag{3.21}$$

1. $c_{33} = cos\theta \Rightarrow sin\theta = \sigma\sqrt{1 - c_{33}^2}$ where $\sigma = \pm 1$
   $\theta = atan2(\sigma\sqrt{1 - c_{33}^2}, c_{33})$
   $\sigma = 1$ leads to θ is acute angle and $\sigma = -1$ leads to θ is obtuse angle. Generally, $\sigma = 1$ is preferred.

2. $\psi = atan2(c_{13}, c_{23})$

3. $\phi = atan2(c_{31}, -c_{32})$

3.3.3 Quaternion to DCM

Quaternion to DCM transformation can be performed using Eq. 3.22, [40].

$$\begin{aligned} A(\boldsymbol{q}) &= (q_4^2 - \|\boldsymbol{q}_{1:3}\|^2)\boldsymbol{I}_3 - 2q_4[\boldsymbol{q}_{1:3}\times] + 2\boldsymbol{q}_{1:3}\boldsymbol{q}_{1:3}^T \\ &= \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_2q_1 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_3q_1 + q_2q_4) & 2(q_3q_2 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \end{aligned} \tag{3.22}$$

3.3.4 DCM to Quaternion

In this subsection, the process of extracting a quaternion from a DCM is discussed. To achieve this, a robust method known as Shepperd's algorithm is utilized. However, a modified version of Shepperd's algorithm is employed in this thesis. Since the details of this algorithm are beyond the scope of this context, readers can refer to the relevant reference for further information, [39].

3.3.5 Euler Angles to Quaternion

Transformation from euler angles to quaternion can be done in two steps: first, transform euler angles to DCM using Eq. 3.17, and then transform DCM to quaternion by using Shepperd's algorithm.

3.3.6 Quaternion to Euler Angles

Quaternion to Euler angles transformation can be done similarly to Euler angles to quaternion by two steps. Firstly, the transformation from quaternion to DCM can be performed using Eq. 3.22, and then Euler angles can be extracted using Eq. 3.20 or Eq. 3.21.

3.4 Attitude Kinematics

The kinematic is provided in Eq. 3.23.

$$\frac{dp}{dt} = f(p, w) \tag{3.23}$$

where

p = variable which represents the spacecraft attitude
w = angular velocity spacecraft

One can examine the equation for three different attitude representations. While Euler angle representation has only 3 parameters for attitude, the function $f(p, w)$ is transcendental in terms of p. This is due to the fact that trigonometric functions can be represented on computers by their power series expansions. Additionally, as described in Section 3.2.2 , Euler angles have different singularities depending on the type of rotation sequence used. In contrast, the DCM and quaternion representations require 9 and 4 parameters, respectively. Notably, $f(p, w)$ is linear with respect to p when using DCM and quaternion representation. Hence, there are two choices for attitude representation: DCM and quaternion.

In this thesis, quaternions are chosen as the attitude representation method. There are several reasons for this choice, as outlined in [51]:

Advantages of using quaternions for attitude representation

- The function $f(p, w)$ exhibits linearity with respect to p.

- Quaternions use only four parameters to specify attitude, as opposed to the nine parameters required when using DCMs.
- Quaternions are computationally more efficient than DCMs. Quaternions integrate a vector differential equation with less computations [51].
- Satisfying the unit norm constraint for quaternions is easier compared to the re-orthonormalization process required for DCMs [51].

In 3.24 and 3.25, the attitude kinematics equations are provided.

$$\dot{\boldsymbol{q}} = \frac{1}{2}\Omega(\omega)\boldsymbol{q} \tag{3.24}$$

where

$$\Omega(\omega) = \begin{bmatrix} -[\boldsymbol{\omega}\times] & 0 \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix} \tag{3.25}$$

3.5 Attitude Dynamics

The dynamics of the satellite can be described by Euler's rotational equation of motion. Since the inertia matrix is constant in the body-fixed frame, the angular momentum of the spacecraft is taken around the mass center. Therefore, the Coriolis transport theorem is used, resulting in the final equation shown in Eq. 3.26.

$$\boldsymbol{J}\dot{\boldsymbol{\omega}} = -[\boldsymbol{\omega}\times]\boldsymbol{J}\boldsymbol{\omega} + \boldsymbol{L} \tag{3.26}$$

where

$\boldsymbol{J}$ = Inertia matrix in a body frame
$\boldsymbol{L}$ = Moment acting on body about mass center

Eq. 3.26 can be used in situations where reaction wheels are not used, such as when magnetorquers are used. However, when reaction wheels are used, Eq. 3.26 is not suitable because reaction wheels create internal moments that change the spacecraft's

angular momentum since they are part of the body. In this case, a modified equation must be used. Eq. 3.27.

$$J\dot{\omega} = -[\omega\times](J\omega + h) - \dot{h} + L \tag{3.27}$$

where

h = Wheel angular momentum
$\dot{h}$ = Wheel torque

In the case where only the reaction wheel is used, i.e., $L = 0$, an equivalent but more useful form is given by Eq. 3.28, as described in reference [40]. This form is particularly useful for defining control laws for the spacecraft, such as sliding-mode control and reaction wheel saturation at Sections 4.2.2 and 4.2.3.

$$\begin{aligned} J\dot{\omega} &= -[\omega\times]J\omega + \bar{L} \\ \dot{h} &= -[\omega\times]h - \bar{L} \end{aligned} \tag{3.28}$$

where

$\bar{L}$ = Effective wheel torque

Using the satellite attitude kinematics equations, Eq. 3.24 and 3.25, and the satellite attitude dynamics equation, Eq. 3.27, the rotation of the satellite in space can be fully defined.

3.6 Measurement Equations

This subsection presents the measurement equations used to generate the simulated measurements.

- **Gyroscope**

$$\tilde{\omega} = \omega + b_g + \eta \tag{3.29}$$

$\tilde{\boldsymbol{\omega}}$ shows the gyroscope measurements and $\boldsymbol{\omega}$ shows the real angular velocity. $\boldsymbol{b}_g$ represents the gyro bias, and $\boldsymbol{\eta}$ represents Gaussian white noise, as shown in Eq. 3.29. In this thesis, the gyro bias is assumed to be constant and the gyro bias instability is not modeled.

- **Magnetometer**

$$\tilde{\boldsymbol{B}}_B = A_{BI}\boldsymbol{B}_I + \boldsymbol{\eta} \tag{3.30}$$

For a fully calibrated magnetometer as indicated in the reference [25], the magnetometer measurement model can be written as shown in Eq. 3.30. Here, $\boldsymbol{B}_I$ represents the magnetic field in the inertial frame which is ECI, $\boldsymbol{B}_B$ represents the magnetometer measurements in the body frame, and $\boldsymbol{\eta}$ represents Gaussian white noise.

- **Sun Sensor**

$$\tilde{\boldsymbol{S}}_B = A_{BI}\boldsymbol{S}_I + \boldsymbol{\eta} \tag{3.31}$$

Sun sensor measurement model can be written as the magnetometer measurement model, Eq. 3.31. $\boldsymbol{S}_I$ represents the sun direction vector in the inertial frame, $\boldsymbol{S}_B$ represents the sun sensor measurements in the body frame, and $\boldsymbol{\eta}$ represents Gaussian white noise.

- **Star Tracker**

$$\boldsymbol{q}_{str} = \boldsymbol{q}_{noise}^{-1} \otimes \boldsymbol{q} \tag{3.32}$$

In Eq. 3.32, the variable $\boldsymbol{q}_{str}$ represents the star tracker measurement, $\boldsymbol{q}_{noise}$ represents the quaternion noise on the star tracker, and $\boldsymbol{q}$ represents the true value of the quaternions, [25]. The quaternion noise, $\boldsymbol{q}_{noise}$, is modeled as a quaternion error, with the small angle assumption, Eq. 3.33. Additionally, the symbol η_1 represents the Gaussian white noise corresponding to index 1, and the same applies to indices 2 and 3. These noises are assumed to be independent of each other. The symbol ϕ denotes the angle.

$$\boldsymbol{q}_{noise} = \begin{bmatrix} (\phi/2)\eta_1 \\ (\phi/2)\eta_2 \\ (\phi/2)\eta_3 \\ 1 \end{bmatrix} \tag{3.33}$$

3.7 External Torques

In space, various disturbance torques can affect a spacecraft's attitude control, including gravity-gradient torque, magnetic torque, aerodynamic torque, and solar radiation pressure torque [40]. The gravity-gradient torque happens when gravity affects different parts of a spacecraft unequally, leading to a twisting force. Magnetic torque can arise from interaction of the Earth's magnetic field with any current-carrying device. Aerodynamic torque is caused by frictional forces resulting from aerodynamic drag. Solar radiation pressure torque occurs when photons from the sunlight transfer their energy to the spacecraft. These disturbances are important for spacecraft attitude control systems, and designers must take them into account. As our spacecraft is Earth-pointing, the gravity-gradient torque is the primary disturbance torque [40], and therefore, this thesis only models the gravity-gradient torque.

3.7.1 Gravity-Gradient Torque

The gravity-gradient torque is primarily caused by the Earth's varying gravitational forces on the spacecraft's body. This creates a moment around the center of mass. The equation for the gravity-gradient torque, with respect to the center of mass, is provided in the Eq. 3.34. A complete derivation of this equation can be found in [40].

$$\boldsymbol{L}_{gg}^{c} = \frac{3\mu}{r^3}\boldsymbol{n} \times (\boldsymbol{J}^{c}\boldsymbol{n}) \tag{3.34}$$

where

μ = gravitational parameter of the main body which is Earth
r = distance between Earth and center of mass
$\boldsymbol{n}$ = nadir-pointing unit vector
$\boldsymbol{J}^{c}$ = moment of inertia tensor about the center of mass

To resolve this torque, the body axis is used for its simplicity in representing the inertia tensor. In order to calculate the gravity-gradient torque vector $\boldsymbol{L}_{gg}^{c}$ in the body axis, the nadir pointing vector needs to be expressed in terms of the body axis. This is done

using the DCM A_{BO}. The unit vector $\boldsymbol{n}_O$ representing the orbit frame is only along one axis. The Eq. 3.35 is obtained by using these parameters:

$$\boldsymbol{n}_B = A_{BO} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -\sin\theta \\ \cos\theta\sin\psi \\ \cos\theta\cos\psi \end{bmatrix} \tag{3.35}$$

The final form of the Eq. 3.34 can be written as Eq. 3.36. Assuming that the body frame corresponds to the principal axis:

$$\boldsymbol{L}^c_{gg,B} = \frac{3\mu}{r^3} \begin{bmatrix} (\boldsymbol{J}_3 - \boldsymbol{J}_2)\cos^2\theta\cos\psi\sin\psi \\ (\boldsymbol{J}_3 - \boldsymbol{J}_1)\cos\theta\sin\theta\cos\psi \\ (\boldsymbol{J}_1 - \boldsymbol{J}_2)\cos\theta\sin\theta\sin\psi \end{bmatrix} \tag{3.36}$$

CHAPTER 4

ESTIMATION AND CONTROL ALGORITHMS

Missions have distinct phases, and each phase has unique requirements. In a broad sense, these define a mode. Since each mode specifies requirements, the ADCS uses multiple estimation algorithms and controllers. Details regarding these algorithms are provided in this chapter. QUEST and MEKF are two different estimation algorithms. Also, three separate controllers are provided. This consists of the B-dot controller, the sliding-mode controller, and the reaction wheel desaturation controller, also known as the momentum damping controller.

Section 4.1.1 explains the QUEST algorithm. However, before examining the QUEST algorithm, covers TRIAD, Wahba's problem, Davenport's q method, and a complete review of the QUEST.

In Section 4.1.2, detailed information about the MEKF is provided. The section begins with Subsection 4.1.2.1, where two types of MEKF are introduced: MEKF with gyroscope measurements and MEKF with mathematical models. The global and local state definitions are then explained. Next, the filter's components are briefly described, including the measurement update, reset, and propagation parts. In Subsection 4.1.2.2, additive and multiplicative error representations are introduced to provide a more precise explanation of the reset part of the filter. Lastly, in Subsection 4.1.2.3, the measurement update, reset, and propagation parts are presented in detail, including the derivation of the equations.

Section 4.2 of the thesis provides details about the control algorithms used. Details about the B-dot controller are provided in the Subsection 4.2.1. Sliding-mode controller is described at 4.2.2, and reaction wheel desaturation controller is explained at

4.2.3.

4.1 Estimation Algorithms

Spacecraft attitude determination and estimation algorithms can be classified into two categories: static and dynamic. Static attitude determination algorithms calculate the spacecraft's attitude instantaneously whenever the measurements are available, whereas dynamic attitude determination algorithms estimate the attitude as an ongoing process and continuously improving the estimation over time. In this thesis, the QUEST algorithm is utilized for static attitude determination, while the MEKF algorithm is used for dynamic attitude determination. The hybrid algorithm used in this study combines the results obtained from the QUEST algorithm and the MEKF algorithm. Specifically, the results obtained from the QUEST algorithm are used directly in some modes, while in other modes, these results are given into the MEKF as a measurement. Furthermore, different MEKFs are used to estimate various parameters, such as attitude, angular velocity, and gyro bias. The MEKF with gyro measurements estimates the attitude and gyro bias and the MEKF with mathematical models (without gyro measurements) estimates the attitude and angular velocity.

4.1.1 QUEST

QUEST algorithm is a type of static attitude determination algorithm. For any static attitude determination algorithm, it is necessary to have at least two vectors with known resolutions in both the reference frame and the body frame. This is due to the fact that only two pieces of information can be obtained from one vector's resolution in both frames due to the unit-norm constraint. To acquire additional information, a second vector is required, which provides two more pieces of information. However, the system is now over-determined since three attitude coordinates are adequate to specify the attitude in three-dimensional space.

To fully understand the QUEST algorithm, it is important to consider other static attitude determination algorithms as well. These algorithms include the TRIAD (TRIaxial Attitude Determination), Wahba's problem, Davenport's q method, and finally,

QUEST.

- **TRIAD**

  The TRIAD algorithm is one of the earlier published algorithms used for static attitude determination, [59]. In this algorithm, a third coordinate frame, known as the TRIAD frame, is defined in addition to the body and reference frames. The purpose of this intermediate frame is to provide a secondary frame of reference, which eases the finding of the DCM between the reference and body frames, [59]. These frames can be shown as:

  Body frame: $F_B\{\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3\}$

  Reference frame: $F_R\{\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3\}$

  Triad frame: $F_T\{\boldsymbol{t}_1, \boldsymbol{t}_2, \boldsymbol{t}_3\}$

  Assume that two measurements are taken and resolved in the body frame, as well as known in the reference frame, such as the sun direction and magnetic field, respectively. The sun direction can be obtained from the Astronomical Almanac Model, which provides the sun direction in the reference frame resolution, while sun sensor measurement gives the sun direction in the body frame. Similarly, the magnetic field can be obtained from the International Geomagnetic Reference Field (IGRF) in the reference frame, and magnetometer measurements give the magnetic field in the body frame. The sun direction vector is denoted as $\boldsymbol{s}$, and the magnetic field vector is denoted as $\boldsymbol{m}$.

  The first basis vector of the Triad frame, denoted as $\boldsymbol{t}_1$, can be chosen to be the sun direction vector, $\boldsymbol{s}$. The second basis vector, $\boldsymbol{t}_2$, can be chosen as the cross product of the sun direction vector and the magnetic field vector, $\boldsymbol{s} \times \boldsymbol{m}$, which is perpendicular to both vectors. The last basis vector, $\boldsymbol{t}_3$, is chosen to form a right-handed triad with the other two basis vectors.

  In order to determine the transformation matrix from one frame to another, the unit vectors of the frame to be transformed are defined with respect to the transformation frame. As a result, Eq. 4.1 can be expressed as follows:

$$
\begin{aligned}
&\boldsymbol{t}_{1,B} = \boldsymbol{s}_B, \quad \boldsymbol{t}_{1,R} = \boldsymbol{s}_R \\
&\boldsymbol{t}_{2,B} = \big(\boldsymbol{s}_B \times \boldsymbol{m}_S\big)\big/\big(\|\boldsymbol{s}_B \times \boldsymbol{m}_S\|\big), \quad \boldsymbol{t}_{2,R} = \big(\boldsymbol{s}_R \times \boldsymbol{m}_R\big)\big/\big(\|\boldsymbol{s}_R \times \boldsymbol{m}_R\|\big) \\
&\boldsymbol{t}_{3,B} = \boldsymbol{t}_{1,B} \times \boldsymbol{t}_{2,B}, \quad \boldsymbol{t}_{3,R} = \boldsymbol{t}_{1,R} \times \boldsymbol{t}_{2,R}
\end{aligned} \tag{4.1}
$$

After the above procedure, the transformation matrix can be defined as $\hat{\boldsymbol{B}}$ which represents the estimated body frame, Eq. 4.2.

$$
\begin{aligned}
A_{\hat{B}T} &= \begin{bmatrix} \boldsymbol{t}_{1,B} & \boldsymbol{t}_{2,B} & \boldsymbol{t}_{3,B} \end{bmatrix} \\
A_{RT} &= \begin{bmatrix} \boldsymbol{t}_{1,R} & \boldsymbol{t}_{2,R} & \boldsymbol{t}_{3,R} \end{bmatrix}
\end{aligned} \tag{4.2}
$$

Finally, the DCM can be written as:

$$
A_{\hat{B}R} = A_{\hat{B}T} A_{RT}^T \tag{4.3}
$$

- **Wahba's Problem**

  Wahba formulated the attitude determination process as an optimization problem. The cost function can be defined as shown in Eq. 4.4.

$$
L(A) = \frac{1}{2} \sum_{i=1}^{N} a_i ||\mathbf{b}_i - A\mathbf{r}_i||^2 \tag{4.4}
$$

  where

  $\boldsymbol{b}_i$ = Set of N unit vectors measured in a spacecraft's body frame

  $\boldsymbol{r}_i$ = Set of N unit vectors measured in a reference frame

  a_i = Weights

- **Davenport's q Method**

  Davenport transformed the Wahba's problem from the DCM formulation to the quaternion formulation. This involved a modification of the cost function definition presented in the earlier equation, Eq. 4.4.

$$
\begin{aligned}
L(A) &= \frac{1}{2} \sum_{i=1}^{N} a_i \big(\boldsymbol{b}_i - A\boldsymbol{r}_i\big)^T \big(\boldsymbol{b}_i - A\boldsymbol{r}_i\big) \\
&= \frac{1}{2} \sum_{i=1}^{N} a_i \big(\boldsymbol{b}_i^T \boldsymbol{b}_i - \boldsymbol{r}_i^T \boldsymbol{r}_i - 2\boldsymbol{b}_i^T A \boldsymbol{r}_i\big) \\
&= \frac{1}{2} \sum_{i=1}^{N} a_i \big(1 - \boldsymbol{b}_i^T A \boldsymbol{r}_i\big)
\end{aligned} \tag{4.5}
$$

According to Eq. 4.5, the objective is to minimize the value of $L(A)$ by maximizing the value of Eq. 4.6.

$$g = \sum_{i=1}^{N} a_i \boldsymbol{b}_i^T A \boldsymbol{r}_i \tag{4.6}$$

Davenport expressed the g function using quaternions. Further information about the process can be found in [19].

$$g = \boldsymbol{q}^T K(B) \boldsymbol{q} \tag{4.7}$$

where

$$\begin{aligned} K(B) &= \begin{bmatrix} B + B^T - tr(B) I_3 & \boldsymbol{z} \\ \boldsymbol{z}^T & tr(B) \end{bmatrix} \\ B &= \textstyle\sum_{i=1}^{N} a_i \boldsymbol{b}_i \boldsymbol{r}_i^T \\ \boldsymbol{z} &= \begin{bmatrix} B_{23} - B_{32} \\ B_{31} - B_{13} \\ B_{12} - B_{21} \end{bmatrix} \end{aligned}$$

Since quaternions also satisfy the unit norm constraint, a Lagrange multiplier is added to the g function. Then, Eq. 4.7 is rewritten as:

$$g' = \boldsymbol{q}^T K(B) \boldsymbol{q} - \lambda(\boldsymbol{q}^T \boldsymbol{q} - 1) \tag{4.8}$$

To find the maximum of Eq. 4.8, the differentiation of g' with respect to $\boldsymbol{q}$ is taken.

$$\frac{dg'}{d\boldsymbol{q}} = 2K(B)\boldsymbol{q} - 2\lambda\boldsymbol{q} = 0 \tag{4.9}$$

$\Rightarrow$

$$K(B)\boldsymbol{q} = \lambda\boldsymbol{q} \tag{4.10}$$

Based on Eq. 4.10, the problem can be reformulated as an eigenvector-eigenvalue problem. This means that one of the eigenvectors of $K(B)$ is the solution to the

maximization problem. To determine which eigenvector this is, one can use g function in Eq. 4.7.

$$g = \boldsymbol{q}^T K(B)\boldsymbol{q} = \lambda \boldsymbol{q}^T \boldsymbol{q} = \lambda \tag{4.11}$$

As shown in Eq. 4.11, the solution to Wahba's problem is the eigenvector corresponding to the maximum eigenvalue of $K(B)$. The next step is to find the eigenvalues and eigenvectors of the 4×4 matrix.

- **QUEST**

  The QUEST algorithm provides a solution to the problem of finding the eigenvalue and eigenvector of the 4×4 matrix. Shuster noticed that the maximum eigenvalue λ_{max} is very close to the sum of the weights, Eq. 4.12.

$$\begin{aligned} L(A) &= \sum_{i=1}^{N} a_i - g \Rightarrow \\ L(A)^\star &= \sum_{i=1}^{N} a_i - \lambda_{max} \Rightarrow \\ \lambda_{max} &= \sum_{i=1}^{N} a_i - L(A)^\star \Rightarrow \\ \lambda_{max} &\approx \sum_{i=1}^{N} a_i \end{aligned} \tag{4.12}$$

  The maximum eigenvalue, λ_{max}, can be computed using Eq. 4.12. Following this discovery, the Newton-Raphson algorithm can be utilized to find the root of $f(s)$, with the initial guess for the exact eigenvalue Eq. 4.12 can be used. Since the real root is very close to the sum of the weights, the Newton-Raphson algorithm converges easily.

$$f(s) = det(K(B) - sI_4) \tag{4.13}$$

$$\begin{cases} \lambda_0 = \sum_{i=1}^{N} a_i \\ \lambda_1 = \lambda_0 - \frac{f(\lambda_0)}{f'(\lambda_0)} \\ . \\ . \\ . \\ \lambda_{max} = \lambda_i = \lambda_{i-1} - \frac{f(\lambda_{i-1})}{f'(\lambda_{i-1})} \end{cases} \tag{4.14}$$

$$\hat{\mathbf{q}} = \alpha \begin{bmatrix} adj(\rho I_3 - S)z \\ det(\rho I_3 - S) \end{bmatrix} \tag{4.15}$$

where

$S = B + B^T$

$\rho = \lambda_{max} + tr(B)$

$\alpha =$ Normalization constant of $\hat{\mathbf{q}}$.

In contrast to TRIAD, which is a deterministic attitude determination algorithm, QUEST belongs to the class of optimization-based static attitude determination algorithms. In TRIAD, the second basis vector of the transformed magnetic field frame, F_T, is given by the cross product of the sun direction vector, $\boldsymbol{s}$, and the magnetic field vector, $\boldsymbol{m}$. As a result, some information about the magnetic field is lost in the process, making the decision of the first basis vector of F_T crucial. On the other hand, QUEST utilizes all the available information and is capable of finding the optimal solution. Additionally, it is faster than explained static attitude determination algorithms. Therefore, in this thesis, the QUEST algorithm was chosen.

4.1.2 MEKF

This subsection provides a detailed explanation of the MEKF. It begins by introducing two variations of the filter: MEKF with gyroscope measurements and MEKF with mathematical models. The definitions of additive and multiplicative error states are

then given. The subsection further includes the measurement update, reset, and propagation equations of the MEKF. To enhance navigation in the subsection, Fig. 4.1 is provided, which shows the relevant parts and equations.

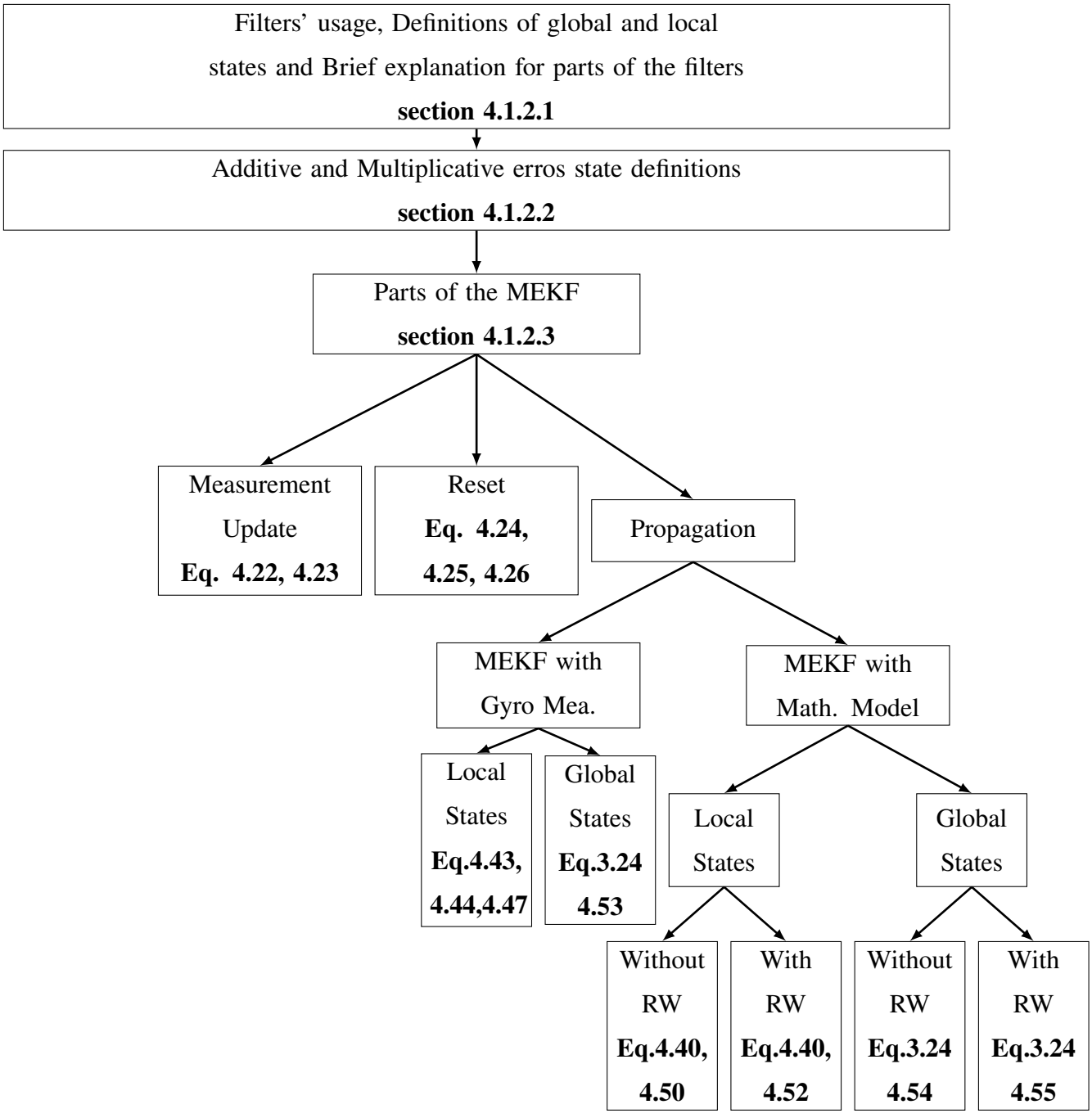


Figure 4.1: Outline for the MEKF

4.1.2.1 Filters Usage, Definition of the States and Filters Parts

- **MEKF with Gyroscope Measurements**

  This filter is called MEKF with gyroscope measurements. It is specifically designed to be used when gyroscope measurements are available. However, it's important to note that gyroscope measurements are not used directly in the measurement update part of the filter. Instead, when gyroscope measurements are available, the filter uses the corrected measurements to propagate the attitude and estimates the gyroscope bias instead of directly estimating the angular velocity. The only measurement source used in this filter is quaternion measurements. In the following sections of the thesis, states of the filter are clearly given.

- **MEKF with Mathematical Model**

  This filter is employed when gyroscope measurements are not available. In this case, instead of estimating the gyroscope bias, the angular velocity is estimated directly using Euler's dynamics. As a result, the filter is referred to as the MEKF with a mathematical model. Similar to the MEKF with gyroscope measurements, the only measurement utilized in this filter is quaternion. The section that follows will go into more detail about the filter states.

As previously stated, gyroscope measurement availability changes the filter states, and if gyroscope measurement is available, it is not used as a measurement source for the filter. In fact, in some situations, gyroscope measurements may not be available during the operation of a spacecraft, or using two redundant algorithms may be desirable. Thus, from a hybrid algorithm perspective, the usage of both filters is important.

- **Global States**

  The nominal state of a filter, which means not an error state, is the state vector that describes the state space of the phenomenon being modeled. Global states are used in the reset and propagation parts of the MEKF. The global states used in both MEKF with gyroscope measurements and mathematical models are listed below:

  - MEKF with gyroscope measurements

    - Quaternion - $\boldsymbol{q}$
    - Gyroscope bias - $\boldsymbol{b}_g$
  - MEKF with mathematical model
    - Quaternion - $\boldsymbol{q}$
    - Angular velocity - $\boldsymbol{\omega}$

- **Local States**

  Local states, also known as error states, are used in the measurement update and reset parts of the filter. In contrast to global states, they represent the deviation of the current state estimate from the nominal state. The global states used in both MEKF with gyroscope measurements and mathematical models are listed below:

  - MEKF with gyroscope measurements
    - Attitude angle error - $\delta\boldsymbol{\theta}$
    - Gyroscope bias error - $\Delta\boldsymbol{b}_g$
  - MEKF with mathematical model
    - Attitude angle error - $\delta\boldsymbol{\theta}$
    - Angular velocity error - $\Delta\boldsymbol{\omega}$

  Utilizing local states in the MEKF has several advantages. Instead of using a 4×4 covariance matrix for the quaternion, using Euler angle error ($\delta\boldsymbol{\theta}$) yields a 3×3 covariance matrix, which is computationally advantageous. Moreover, the covariance matrix of the MEKF has a physical meaning [40]. In the following sections, the physical meaning of $\delta\boldsymbol{\theta}$ will be explained.

Instead of using separate error states for $\Delta\boldsymbol{b}_g$ and $\Delta\boldsymbol{\omega}$ in the some equations, the single error state $\Delta\boldsymbol{\xi}$ can be used, which represents both of them. Therefore, local and global states are written as matrix form in the Eqs. 4.16 and 4.17.

$$\Delta\boldsymbol{x} = \begin{bmatrix} \delta\boldsymbol{\theta} \\ \Delta\boldsymbol{\xi} \end{bmatrix} \tag{4.16}$$

where

$\delta\theta$ = Local states for the attitude representation, Euler angle error

$\Delta\xi$ = Local states for "other variables" to estimated

$$\boldsymbol{x} = \begin{bmatrix} q \\ \xi \end{bmatrix} \tag{4.17}$$

where

q = Global states for the attitude representation, quaternion

ξ = Global states for the "other variables" to estimated

- **Measurement Update**

  The measurement update step in the MEKF updates the local state vector.

- **Reset**

  The reset step involves transferring the updated information from the error state to the global state representation, and then resetting the components of the local state to zero. One possible interpretation of "setting zero" is to view it as the propagation process of the local states in the filter, where all local states are set to zero.

- **Propagation**

  During the propagation step of the MEKF, the global variables are updated and advanced to the next time step. On the other hand, the error state variables do not need to be propagated because they are already zero over the reset step. Therefore, it may be inferred that error states are not utilized in the propagation step.

As previously explained, the filters are divided into three sections and each of these sections is analyzed separately. However, prior to examining these filter sections, the additive and multiplicative errors will be discussed. This part especially important to explain the reset part of the filters.

4.1.2.2 Additive and Multiplicative Error Representation

Error at some quantity can be represented as addition, such as error at position or error at velocity. As shown in Eq. 4.18, let $\boldsymbol{\xi}$ be a certain quantity. The true value of $\boldsymbol{\xi}$ can be obtained by adding the estimated value $\hat{\boldsymbol{\xi}}$ and the error value $\Delta\boldsymbol{\xi}$. This additive representation is used in the filter's reset stage to update the estimation of "other variables", such as $\boldsymbol{b}_g$ and $\boldsymbol{\omega}$. Attitude error can also be represented in this way. However, there is an alternative way of representing attitude error multiplicatively, as shown in Eq. 4.19. The equation represents the true DCM from the reference frame to the body as A_{BR}. The estimated DCM from the reference frame to the estimated body frame is denoted as $A_{\hat{B}R}$. The error between the estimated and true body frames is represented by $A_{B\hat{B}}$, which is the DCM from the estimated to the true body frame.

$$\boldsymbol{\xi} = \hat{\boldsymbol{\xi}} + \Delta\boldsymbol{\xi} \tag{4.18}$$

where

$\boldsymbol{\xi}$ = True value of $\boldsymbol{\xi}$
$\hat{\boldsymbol{\xi}}$ = Estimated value of $\boldsymbol{\xi}$
$\Delta\boldsymbol{\xi}$ = Error on the $\boldsymbol{\xi}$

$$A_{BR} = A_{B\hat{B}} A_{\hat{B}R} \tag{4.19}$$

where

A_{BR} = True DCM from reference to body frame
$A_{\hat{B}R}$ = Estimated DCM from reference to estimated body frame
$A_{B\hat{B}}$ = Error from estimated to real body frame

The MEKF reset part, quaternion measurement update can be done in both ways, either additively or multiplicatively. However, there are some disadvantages of using the additive quaternion error update. Quaternion update may not obey the normalization constraint, which can create biases in the filter and cause ill-conditioned covariance matrices [40]. It is important to note that the normalization of the quaternion cannot

be done during the update part, as it is a mathematical constraint that must be satisfied during the propagation part of the filter.

Eq. 4.19 can also be represented as Eq. 4.20 in terms of quaternions.

$$\boldsymbol{q} = \delta\boldsymbol{q}(\delta\boldsymbol{\theta}) \otimes \hat{\boldsymbol{q}} \tag{4.20}$$

where

$\boldsymbol{q}$ = True value of quaternion
$\hat{\boldsymbol{q}}$ = Estimated value of quaternion
$\delta\boldsymbol{q}(\delta\boldsymbol{\theta})$ = Error on the quaternion

As stated in Equation 4.16, $\delta\boldsymbol{\theta}$ is a local attitude representation that represents Euler angle error. The relationship between Euler angle error and quaternion error can be found in Equation 4.21. For further information, [17, 40] is a suitable resource.

$$\delta\boldsymbol{\theta} = \delta\phi\boldsymbol{e}_1 + \delta\theta\boldsymbol{e}_2 + \delta\psi\boldsymbol{e}_3 = 2\delta\boldsymbol{q}_{1:3} \tag{4.21}$$

The MEKF algorithm utilizes a quaternion error representation that is multiplicatively updated, leading to its name as the Multiplicative Extended Kalman Filter.

4.1.2.3 Parts of the MEKF

This section provides a detailed examination of the measurement, reset, and propagation parts of the filters, including a derivation of the equations. In this section, all updated and propagated values are denoted with a hat symbol (ˆ). The primary objective of this section is to provide a comprehensive examination of the filter and to present the derivation of the equations. As a result, the distinction between updated and propagated states is not always clear. To summarize the discussion, algorithms for both filters are provided, and at this point, the updated and propagated values are distinguishable.

Measurement Update The measurement update step is the same for both the MEKF with gyroscope measurement and mathematical model filters. Both filters use quaternions as the measurement, which may come from the star tracker or the QUEST algorithm. This quaternion value is not used directly in the filter; instead, it is used to calculate the measured Euler angle error. Therefore, the only measured value is $\delta\tilde{\boldsymbol{\theta}}$, obtained from the measured quaternion $\tilde{\boldsymbol{q}}$ in Eq. 4.22. To get $\delta\tilde{\boldsymbol{\theta}}$ the Rodrigues vector parametrization is used. This is important because there are two possible representations for a given rotation with quaternions, $\boldsymbol{q}$ and $-\boldsymbol{q}$. The sign ambiguity can cause problems when comparing or combining multiple rotations [40]. Furthermore, the observation matrix is defined in Eq. 4.23 since only measured local state is Euler angle error.

$$\boldsymbol{y} = \delta\tilde{\boldsymbol{\theta}} = 2\frac{(\tilde{\boldsymbol{q}} \otimes \hat{\boldsymbol{q}}^{-1})_{1:3}}{(\tilde{\boldsymbol{q}} \otimes \hat{\boldsymbol{q}}^{-1})_4} \tag{4.22}$$

where

$\delta\tilde{\boldsymbol{\theta}}$ = Measured Euler angle
$\tilde{\boldsymbol{q}}$ = Measured quaternion
$\hat{\boldsymbol{q}}$ = Estimated quaternion

$$H = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{4.23}$$

where

H = Observability matrix

Reset Reset is the process of transferring information between local and global states, which is explained in Eqs. 4.24 and 4.25. As mentioned earlier, error in Euler angles are denoted as $\delta\hat{\boldsymbol{\theta}}$ and they are transferred to the global state represented by the quaternion $\hat{\boldsymbol{q}}$ in a multiplicative manner. However, when it comes to "other variables" that need to be estimated, the information transformation is done additively.

$$\begin{aligned}\hat{\boldsymbol{q}} &= \delta \boldsymbol{q}(\delta\hat{\boldsymbol{\theta}}) \otimes \hat{\boldsymbol{q}} \\ &= \frac{1}{\sqrt{1 + \|\delta\hat{\boldsymbol{\theta}}/2\|^2}} \begin{bmatrix} \delta\hat{\boldsymbol{\theta}}/2 \\ 1 \end{bmatrix} \otimes \hat{\boldsymbol{q}}\end{aligned} \tag{4.24}$$

$$\hat{\boldsymbol{\xi}} = \hat{\boldsymbol{\xi}} + \Delta\hat{\boldsymbol{\xi}} \tag{4.25}$$

Another important aspect of the reset section is that it involves propagating the local variables and setting the time propagation of the local variable to a zero vector. This propagation refers to time propagation, meaning that in the propagation section of the filter, only global variables are propagated.

$$\begin{aligned}\delta\hat{\boldsymbol{\theta}} &= 0 \\ \Delta\hat{\boldsymbol{\xi}} &= 0\end{aligned} \tag{4.26}$$

Propagation In this section, we will examine the propagation step separately for two types of filters: the MEKF with gyroscope measurements and the mathematical model filter. Firstly, we will present the mathematical model for the local states, which is important for the propagation of the covariance. Secondly, we will discuss the propagation part of the global variables state. Also, in the derivation of the equations the references [17, 40] are used.

- **Local States**

  Since the Euler angle error $\delta\boldsymbol{\theta}$ is common to both filters, we will begin by presenting the derivation of $\delta\dot{\boldsymbol{\theta}}$ for both filters.

  Eq. 4.20 can be expressed as:

$$\delta \boldsymbol{q} = \boldsymbol{q} \otimes \hat{\boldsymbol{q}}^{-1} \tag{4.27}$$

  Taking the derivative of Eq. 4.27, we get:

$$\delta\dot{\boldsymbol{q}} = \dot{\boldsymbol{q}} \otimes \hat{\boldsymbol{q}}^{-1} + \boldsymbol{q} \otimes \dot{\hat{\boldsymbol{q}}}^{-1} \tag{4.28}$$

To obtain the derivative of the estimated quaternion inverse, denoted as $\dot{\hat{\boldsymbol{q}}}^{-1}$, the following derivation is presented.

The quaternion kinematic equation is given as follows:

$$\dot{\hat{\boldsymbol{q}}} = \frac{1}{2}\Omega(\hat{\boldsymbol{\omega}})\hat{\boldsymbol{q}} \tag{4.29}$$

Also, it is known that:

$$\hat{\boldsymbol{q}} \otimes \hat{\boldsymbol{q}}^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T \tag{4.30}$$

Taking the time derivative of Eq. 4.30 yields:

$$\dot{\hat{\boldsymbol{q}}} \otimes \hat{\boldsymbol{q}}^{-1} + \hat{\boldsymbol{q}} \otimes \dot{\hat{\boldsymbol{q}}}^{-1} = 0 \tag{4.31}$$

Substituting Eq. 4.29 into Eq. 4.31, we get:

$$\frac{1}{2}\Omega(\hat{\boldsymbol{\omega}})\hat{\boldsymbol{q}} \otimes \hat{\boldsymbol{q}}^{-1} + \hat{\boldsymbol{q}} \otimes \dot{\hat{\boldsymbol{q}}}^{-1} = 0 \tag{4.32}$$

The equation 4.32 can be expressed as:

$$\frac{1}{2}\begin{bmatrix} \hat{\boldsymbol{\omega}} \\ 0 \end{bmatrix} + \hat{\boldsymbol{q}} \otimes \dot{\hat{\boldsymbol{q}}}^{-1} = 0 \tag{4.33}$$

$\Rightarrow$

$$\dot{\hat{\boldsymbol{q}}}^{-1} = -\frac{1}{2}\hat{\boldsymbol{q}}^{-1} \otimes \begin{bmatrix} \hat{\boldsymbol{\omega}} \\ 0 \end{bmatrix} \tag{4.34}$$

Furthermore, it is important to note that:

$$\dot{\boldsymbol{q}} = \frac{1}{2}\Omega(\boldsymbol{\omega})\boldsymbol{q} = \frac{1}{2}\begin{bmatrix} \boldsymbol{\omega} \\ 0 \end{bmatrix} \otimes \boldsymbol{q} \tag{4.35}$$

Therefore, we can rewrite Eq. 4.28 using Eqs. 4.34 and 4.35 as follows:

$$\delta\dot{\boldsymbol{q}} = \frac{1}{2}\left\{ \begin{bmatrix} \boldsymbol{\omega} \\ 0 \end{bmatrix} \otimes \delta\boldsymbol{q} - \delta\boldsymbol{q} \otimes \begin{bmatrix} \hat{\boldsymbol{\omega}} \\ 0 \end{bmatrix} \right\} \tag{4.36}$$

After some manipulation of Eq. 4.36, Eq. 4.37 can be written as given in [17]:

$$\delta\dot{\boldsymbol{q}} = -\begin{bmatrix} [\hat{\boldsymbol{\omega}}\times]\delta\boldsymbol{q}_{1:3} \\ 0 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} \Delta\boldsymbol{\omega} \\ 0 \end{bmatrix} \otimes \delta\boldsymbol{q} \tag{4.37}$$

In Eq. 4.37, a non-linear term appears in the last term. To simplify this term, a first-order approximation can be used, as shown in Eq. 4.38.

$$\frac{1}{2}\begin{bmatrix} \Delta\boldsymbol{\omega} \\ 0 \end{bmatrix} \otimes \delta\boldsymbol{q} \approx \frac{1}{2}\begin{bmatrix} \Delta\boldsymbol{\omega} \\ 0 \end{bmatrix} \tag{4.38}$$

When using Eq. 4.38, Eq. 4.37 can be expressed as follows:

$$\begin{aligned} \delta\dot{\boldsymbol{q}}_{1:3} &= -[\hat{\boldsymbol{\omega}}\times]\delta\boldsymbol{q}_{1:3} + \frac{1}{2}\Delta\boldsymbol{\omega} \\ \delta\dot{q}_4 &= 0 \end{aligned} \tag{4.39}$$

By using the equation 4.21, we can obtain equation 4.40, which expresses the propagation of $\delta\boldsymbol{\theta}$ over time in terms of $\Delta\boldsymbol{\omega}$. It is important for the MEKF with mathematical model.

$$\begin{aligned} \frac{1}{2}\delta\dot{\boldsymbol{\theta}}_{1:3} &= -\frac{1}{2}[\hat{\boldsymbol{\omega}}\times]\delta\boldsymbol{\theta}_{1:3} + \frac{1}{2}\Delta\boldsymbol{\omega} \\ \delta\dot{\boldsymbol{\theta}}_{1:3} &= -[\hat{\boldsymbol{\omega}}\times]\delta\boldsymbol{\theta}_{1:3} + \Delta\boldsymbol{\omega} \end{aligned} \tag{4.40}$$

Now we need to express $\Delta\boldsymbol{\omega}$ in terms of $\Delta\boldsymbol{b}_g$. By using Eq. 3.29, we can write Eq. 4.41 as follows:

$$\begin{aligned} \Delta\boldsymbol{\omega} &= \boldsymbol{\omega} - \hat{\boldsymbol{\omega}} \\ &= \tilde{\omega} - \boldsymbol{b}_g - \boldsymbol{\eta} - (\tilde{\boldsymbol{\omega}} - \hat{\boldsymbol{b}}_g) \\ &= (\hat{\boldsymbol{b}}_g - \boldsymbol{b}_g) - \boldsymbol{\eta} \\ &= -(\Delta\boldsymbol{b}_g + \boldsymbol{\eta}) \end{aligned} \tag{4.41}$$

Inserting Eq. 4.41 into Eq. 4.39, we have:

$$\delta\dot{\boldsymbol{q}}_{1:3} = -[\hat{\boldsymbol{\omega}}\times]\delta\boldsymbol{q}_{1:3} - \frac{1}{2}(\Delta\boldsymbol{b}_g + \boldsymbol{\eta}) \tag{4.42}$$

Using the equation 4.21, we can write Eq. 4.42 as:

$$\begin{aligned} \frac{1}{2}\delta\dot{\boldsymbol{\theta}} &= -\frac{1}{2}[\hat{\boldsymbol{\omega}}\times]\delta\boldsymbol{\theta} - \frac{1}{2}(\Delta\boldsymbol{b}_g + \boldsymbol{\eta}_{gn}) \\ \delta\dot{\boldsymbol{\theta}} &= -[\hat{\boldsymbol{\omega}}\times]\delta\boldsymbol{\theta} - (\Delta\boldsymbol{b}_g + \boldsymbol{\eta}_{gn}) \end{aligned} \tag{4.43}$$

Eq. 4.43 is important for the MEKF with gyroscope measurement filter as it gives the propagation of $\delta\boldsymbol{\theta}$ in terms of the gyroscope bias $\Delta\boldsymbol{b}_g$.

The remaining parts to be derived are the propagation equations for $\Delta\boldsymbol{b}_g$ and $\Delta\boldsymbol{\omega}$, which will be derived separately.

– **Local States of the MEKF with Gyroscope Measurements**

As shown in Eq. 4.43, $\delta\dot{\boldsymbol{\theta}}$ can be expressed in terms of $\delta\boldsymbol{\theta}$ and $\Delta\boldsymbol{b}_g$. Therefore, in this part, only the time propagation of $\Delta\boldsymbol{b}_g$ will be derived.

In the filter bias error, $\Delta\hat{\boldsymbol{b}}_g$ is taken as constant:

$$\Delta\dot{\hat{\boldsymbol{b}}}_g = 0 \tag{4.44}$$

Therefore, for the MEKF with gyroscope measurement mathematical model of the error state can be given as:

$$\begin{bmatrix} \delta\dot{\boldsymbol{\theta}} \\ \Delta\dot{\boldsymbol{b}}_g \end{bmatrix} = F(t) \begin{bmatrix} \delta\boldsymbol{\theta} \\ \Delta\boldsymbol{b}_g \end{bmatrix} + G(t)\boldsymbol{w}(\boldsymbol{t}) \tag{4.45}$$

$$\begin{bmatrix} \delta\dot{\boldsymbol{\theta}} \\ \Delta\dot{\boldsymbol{b}}_g \end{bmatrix} = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}\times] & -I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \begin{bmatrix} \delta\boldsymbol{\theta} \\ \Delta\boldsymbol{b}_g \end{bmatrix} + \begin{bmatrix} -I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} \end{bmatrix} \boldsymbol{w}(\boldsymbol{t}) \tag{4.46}$$

However, since the Kalman filter is in the discrete form, the state-driven matrix can be written approximately as shown in Equation 4.47:

$$F_k \approx \begin{bmatrix} -[\hat{\boldsymbol{\omega}}\times] & -I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \Delta t + I_{6\times6} \tag{4.47}$$

where:

$\Delta t =$ Time incrementation

– **Local States of the MEKF with Mathematical Model**

The Eq. 4.40 represents the relationship between $\delta\dot{\boldsymbol{\theta}}$ and $\Delta\boldsymbol{\omega}$. In this section, the time propagation of $\Delta\dot{\boldsymbol{\omega}}$ is derived. The name of the filter, MEKF with mathematical model, suggests that $\Delta\dot{\boldsymbol{\omega}}$ is obtained using the mathematical model, which is based on the Euler rotation formula.

To obtain $\Delta\dot{\boldsymbol{\omega}}$ using the Euler rotation formula, Eqs. 3.26 and 3.27 are utilized. However, as previously discussed in Section 3.5, the Euler rotation formula can be expressed in two different forms depending on whether a reaction wheel is used or not, which determines whether an internal torque exists or not.

Therefore, the mathematical model of this filter is also formulated in two forms:

$$\begin{bmatrix} \delta\dot{\boldsymbol{\theta}} \\ \Delta\dot{\boldsymbol{\omega}} \end{bmatrix} = F(t) \begin{bmatrix} \delta\boldsymbol{\theta} \\ \Delta\boldsymbol{\omega} \end{bmatrix} + G(t)\boldsymbol{w(t)} \tag{4.48}$$

where:

$$G(t) = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} \end{bmatrix}$$

∗ **without reaction wheel**

$$F(t) = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}\times] & I_{3\times3} \\ 0_{3\times3} & J^{-1}([J\hat{\boldsymbol{\omega}}\times] - [\hat{\boldsymbol{\omega}}\times]J) \end{bmatrix} \tag{4.49}$$

$\Rightarrow$

$$F_k \approx \begin{bmatrix} -[\hat{\boldsymbol{\omega}}\times] & I_{3\times3} \\ 0_{3\times3} & J^{-1}([J\hat{\boldsymbol{\omega}}\times] - [\hat{\boldsymbol{\omega}}\times]J) \end{bmatrix} \Delta t + I_{6\times6} \tag{4.50}$$

∗ **with reaction wheel**

$$F(t) = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}\times] & I_{3\times3} \\ 0_{3\times3} & J^{-1}([(\boldsymbol{h} + J\hat{\boldsymbol{\omega}})\times] - [\hat{\boldsymbol{\omega}}\times]J) \end{bmatrix} \tag{4.51}$$

$\Rightarrow$

$$F_k \approx \begin{bmatrix} -[\hat{\boldsymbol{\omega}}\times] & I_{3\times3} \\ 0_{3\times3} & J^{-1}([(\boldsymbol{h} + J\hat{\boldsymbol{\omega}})\times] - [\hat{\boldsymbol{\omega}}\times]J) \end{bmatrix} \Delta t + I_{6\times6} \tag{4.52}$$

- **Global States**

  Both the MEKF with gyroscope measurements and the mathematical model filter share the global state $\boldsymbol{q}$, which is propagated using the equation in Eq. 3.24. However, the unshared global states differ between the two filters. In the MEKF with gyroscope measurements, the global state $\boldsymbol{b}_g$ is used, while in the MEKF with mathematical model, $\boldsymbol{\omega}$ is employed as the global state. The derivation of both global states which are uncommon is presented separately for each filter.

  - **Global States of the MEKF with Gyroscope Measurement**

    The gyroscope bias in this filter is assumed to be constant during time propagation, thus ignoring any potential bias instability. As a result, the equation can be expressed as:

    $$\dot{\hat{\boldsymbol{b}}}_g = 0 \tag{4.53}$$

  - **Global States of the MEKF with Mathematical Model**

    The propagation of $\hat{\boldsymbol{\omega}}$ can be obtained using Eqs. 3.26 and 3.27. As mentioned earlier, the satellite attitude dynamics in the local state propagation part varies depending on whether the reaction wheel is utilized in the system. If the reaction wheel is not employed, then Eq. 4.54 can be expressed as follows:

    $$\dot{\hat{\boldsymbol{\omega}}} = \boldsymbol{J}^{-1}(-[\hat{\boldsymbol{\omega}}\times]\boldsymbol{J}\hat{\boldsymbol{\omega}} + \boldsymbol{L}) \tag{4.54}$$

    If the system uses a reaction wheel, then the Eq. 4.55 can be expressed as:

    $$\dot{\hat{\boldsymbol{\omega}}} = \boldsymbol{J}^{-1}(-[\hat{\boldsymbol{\omega}}\times](\boldsymbol{J}\hat{\boldsymbol{\omega}} + \boldsymbol{h}) - \dot{\boldsymbol{h}} + \boldsymbol{L}) \tag{4.55}$$

In conclusion to Section 4.1.2, algorithms 1 and 2 are presented as the final results for both filters.

Algorithm 1 MEKF with gyroscope measurements

$$\boldsymbol{x}_{k+1} = F\boldsymbol{x}_k + G\boldsymbol{w}_k$$

$$\boldsymbol{y}_k = H\boldsymbol{x}_k + \boldsymbol{v}_k$$

$$\boldsymbol{x}_0 \sim N(\boldsymbol{x}_0; \bar{\boldsymbol{x}}_0, \Sigma_0)$$

$$\boldsymbol{w}_k \sim N(\boldsymbol{w}_k; 0, Q)$$

$$\boldsymbol{v}_k \sim N(\boldsymbol{v}_k; 0, R)$$

$\boldsymbol{w}_k$ = process noise

$\boldsymbol{v}_k$ = measurement noise

$G_k = \begin{bmatrix} -I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} \end{bmatrix}$

$H_k = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \end{bmatrix}$

For $k = 0, ..., N$

% Measurement Update

$\hat{\boldsymbol{y}}_{k|k-1} = \delta\hat{\boldsymbol{\theta}}_{k|k-1} = 0$

$\boldsymbol{y}_k = \delta\tilde{\boldsymbol{\theta}}_k = 2\frac{(\tilde{\boldsymbol{q}}_k \otimes \hat{\boldsymbol{q}}^{-1}_{k|k-1})_{1:3}}{(\tilde{\boldsymbol{q}}_k \otimes \hat{\boldsymbol{q}}^{-1}_{k|k-1})_4}$

$S_k = H\Sigma_{k|k-1}H^T + R$

$K_k = \Sigma_{k|k-1}H^T S_k^{-1}$

$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + K_k(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1}) = \begin{bmatrix} \delta\hat{\boldsymbol{\theta}}_{k|k} \\ \Delta\hat{\boldsymbol{b}}_{g,k|k} \end{bmatrix}$

$\Sigma_{k|k} = \Sigma_{k|k-1} - K_k S_k K_k^T$

% Reset

$\hat{\boldsymbol{q}}_{k|k} = \delta\boldsymbol{q}(\delta\hat{\boldsymbol{\theta}}_{\boldsymbol{k|k}}) \otimes \hat{\boldsymbol{q}}_{k|k-1} = \frac{1}{\sqrt{1+||\delta\hat{\boldsymbol{\theta}}_{k|k}/2||^2}} \begin{bmatrix} \delta\hat{\boldsymbol{\theta}}_{k|k}/2 \\ 1 \end{bmatrix} \otimes \hat{\boldsymbol{q}}_{k|k-1}$

$\hat{\boldsymbol{b}}_{g,k|k} = \hat{\boldsymbol{b}}_{g,k|k-1} + \Delta\hat{\boldsymbol{b}}_{g,k|k}$

$\delta\hat{\boldsymbol{\theta}}_{k+1|k} = 0 \quad \& \quad \Delta\hat{\boldsymbol{b}}_{g,k+1|k} = 0$

% Propagation

$\hat{\boldsymbol{\omega}}_{k|k} = \tilde{\boldsymbol{\omega}}_k - \hat{\boldsymbol{b}}_{g,k|k}$

$\hat{\Omega}_k = \hat{\Omega}(\hat{\boldsymbol{\omega}}_{k|k})$

$\hat{\boldsymbol{q}}_{k+1|k} = (I_{4\times4} + \hat{\Omega}_k\Delta t/2)\hat{\boldsymbol{q}}_{k|k}$

$\hat{\boldsymbol{b}}_{g,k+1|k} = \hat{\boldsymbol{b}}_{g,k|k}$

$$F_k = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}_{k|k}\times] & -I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \Delta t + I_{6\times6}$$

$$\Sigma_{k+1|k} = F_k \Sigma_{k|k} F_k^T + GQG^T$$

End For

Algorithm 2 MEKF with mathematical model

$$\boldsymbol{x}_{k+1} = F\boldsymbol{x}_k + G\boldsymbol{w}_k$$

$$\boldsymbol{y}_k = H\boldsymbol{x}_k + \boldsymbol{v}_k$$

$$\boldsymbol{x}_0 \sim N(\boldsymbol{x}_0; \bar{\boldsymbol{x}}_0, \Sigma_0)$$

$$\boldsymbol{w}_k \sim N(\boldsymbol{w}_k; 0, Q)$$

$$\boldsymbol{v}_k \sim N(\boldsymbol{v}_k; 0, R)$$

$\boldsymbol{w}_k$ = process noise

$\boldsymbol{v}_k$ = measurement noise

$G_k = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} \end{bmatrix}$

$H_k = \begin{bmatrix} I_3 & 0_{3\times3} \end{bmatrix}$

For $k = 0, ..., N$

% Measurement Update

$\hat{\boldsymbol{y}}_{k|k-1} = \delta\hat{\boldsymbol{\theta}}_{k|k-1} = 0$

$\boldsymbol{y}_k = \delta\tilde{\boldsymbol{\theta}}_k = 2\frac{(\tilde{\boldsymbol{q}}_k \otimes \hat{\boldsymbol{q}}^{-1}_{k|k-1})_{1:3}}{(\tilde{\boldsymbol{q}}_k \otimes \hat{\boldsymbol{q}}^{-1}_{k|k-1})_4}$

$S_k = H\Sigma_{k|k-1}H^T + R$

$K_k = \Sigma_{k|k-1}H^T S_k^{-1}$

$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + K_k(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1}) = \begin{bmatrix} \delta\hat{\boldsymbol{\theta}}_{k|k} \\ \Delta\hat{\boldsymbol{\omega}}_{k|k} \end{bmatrix}$

$\Sigma_{k|k} = \Sigma_{k|k-1} - K_k S_k K_k^T$

% Reset

$\hat{\boldsymbol{q}}_{k|k} = \delta\boldsymbol{q}(\delta\hat{\boldsymbol{\theta}}_{\boldsymbol{k|k}}) \otimes \hat{\boldsymbol{q}}_{k|k-1} = \frac{1}{\sqrt{1+||\delta\hat{\boldsymbol{\theta}}_{k|k}/2||^2}} \begin{bmatrix} \delta\hat{\boldsymbol{\theta}}_{k|k}/2 \\ 1 \end{bmatrix} \otimes \hat{\boldsymbol{q}}_{k|k-1}$

$\hat{\boldsymbol{\omega}}_{k|k} = \hat{\boldsymbol{\omega}}_{k|k-1} + \Delta\hat{\boldsymbol{\omega}}_{k|k}$

$\delta\hat{\boldsymbol{\theta}}_{k+1|k} = 0 \quad \& \quad \Delta\hat{\boldsymbol{\omega}}_{k+1|k} = 0$

% Propagation

$\hat{\Omega}_k = \hat{\Omega}(\hat{\boldsymbol{\omega}}_{k|k})$

$\hat{\boldsymbol{q}}_{k+1|k} = (I_{4\times4} + \hat{\Omega}_k\Delta t/2)\hat{\boldsymbol{q}}_{k|k}$

$\hat{\boldsymbol{\omega}}_{k+1|k} = \hat{\boldsymbol{\omega}}_{k|k} + \boldsymbol{J}^{-1}(-[\hat{\boldsymbol{\omega}}_{k|k}\times]\boldsymbol{J}\hat{\boldsymbol{\omega}}_{k|k} + \boldsymbol{L})\Delta t$ OR $\hat{\boldsymbol{\omega}}_{k+1|k} = \hat{\boldsymbol{\omega}}_{k|k} + \boldsymbol{J}^{-1}(-[\hat{\boldsymbol{\omega}}_{k|k}\times](\boldsymbol{J}\hat{\boldsymbol{\omega}}_{k|k} + \boldsymbol{h}) - \dot{\boldsymbol{h}} + \boldsymbol{L})\Delta t$

$$F_k = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}_{k|k}\times] & I_{3\times3} \\ 0_{3\times3} & \boldsymbol{J}^{-1}([\boldsymbol{J}\hat{\boldsymbol{\omega}}_{k|k}\times] - [\hat{\boldsymbol{\omega}}_{k|k}\times]\boldsymbol{J}) \end{bmatrix} \Delta t + I_{6\times6} \text{ OR}$$

$$F_k = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}_{k|k}\times] & I_{3\times3} \\ 0_{3\times3} & \boldsymbol{J}^{-1}([(\boldsymbol{h} + \boldsymbol{J}\hat{\boldsymbol{\omega}}_{k|k})\times] - [\hat{\boldsymbol{\omega}}_{k|k}\times]\boldsymbol{J}) \end{bmatrix} \Delta t + I_{6\times6}$$

$\Sigma_{k+1|k} = F_k \Sigma_{k|k} F_k^T + GQG^T$

End For

4.2 Control Algorithms

In this part, the controllers B-dot controller, sliding-mode controller, and reaction wheel desaturation controller, also known as momentum damping controller, are given.

4.2.1 B-dot Controller

The B-dot controller is employed for detumbling the spacecraft, which refers to reducing the angular velocity of the spacecraft. When the spacecraft is released from the rocket, it has a random angular velocity. To activate the estimation and fine-tuning controller algorithm, the spacecraft's angular velocity needs to be brought close to zero. The B-dot algorithm is used for this purpose. The algorithm uses the Earth's magnetic field and the magnetic dipole moment generated by the magnetorquer. Although other actuators, such as thrusters, can also be used, the magnetorquer is used in this thesis.

The magnetic dipole moment created by the magnetorquers can be expressed using Eq. 4.56.

$$\boldsymbol{m} = \frac{k}{\|\boldsymbol{B}\|}\boldsymbol{\omega} \times \boldsymbol{b} \tag{4.56}$$

where

k = a positive scalar gain

$\boldsymbol{B}$ = the Earth's magnetic field

$\boldsymbol{b}$ = the unit vector of the Earth's magnetic field, given by $\boldsymbol{B}/\|\boldsymbol{B}\|$

$\boldsymbol{\omega}$ = the angular velocity of the spacecraft

The interaction between the magnetorquer and Earth's magnetic field leads to torques, which can be described by Equation 4.57.

$$\boldsymbol{L} = \boldsymbol{m} \times \boldsymbol{B} \tag{4.57}$$

To obtain the expression for $\boldsymbol{L}$, one can use Equations 4.56 and 4.57, resulting in:

$$\boldsymbol{L} = \frac{k}{\|\boldsymbol{B}\|}(\boldsymbol{\omega} \times \boldsymbol{b}) \times \boldsymbol{B} = k(\boldsymbol{\omega} \times \boldsymbol{b}) \times \boldsymbol{b} \tag{4.58}$$

Eq. 4.58 assumes that gyroscope measurements are available during the detumble mode. However, in practical situations, some subsystems may not be available at this stage. To address this limitation, the time derivative of the magnetic field vector $\dot{\boldsymbol{B}}$ is used as a substitute for the angular velocity vector $\boldsymbol{\omega}$. This representation is more general.

$$\begin{aligned} D_R\boldsymbol{B} &= D_B\boldsymbol{B} + \boldsymbol{\omega} \times \boldsymbol{B} \\ D_R\boldsymbol{B}_B &= D_B\boldsymbol{B}_B + \boldsymbol{\omega}_B \times \boldsymbol{B}_B \\ &= \dot{\boldsymbol{B}} + \boldsymbol{\omega}_B \times \boldsymbol{B}_B \end{aligned} \tag{4.59}$$

where

$D_R\boldsymbol{B}$ = Time derivative of the magnetic field with respect to reference frame
$D_B\boldsymbol{B}$ = Time derivative of the magnetic field with respect to body frame

As the magnetic field of the Earth is measured by the magnetometer in the body frame, the Coriolis transport theorem can be utilized to determine the magnetic field derivative in the reference frame, which in our case is ECI. Additionally, the assumption $D_R\boldsymbol{B} \ll D_B\boldsymbol{B}$ can be made in Eq. 4.59. Based on this assumption:

$$\dot{\boldsymbol{B}} = -[\boldsymbol{\omega}_B\times]\boldsymbol{B}_B \tag{4.60}$$

If Eq. 4.60 is inserted into Eq. 4.56, the resulting expression is:

$$\begin{aligned} \boldsymbol{m}_B &= \frac{k}{\|\boldsymbol{B}_B\|}\boldsymbol{\omega}_B \times \frac{\boldsymbol{B}_B}{\|\boldsymbol{B}_B\|} \\ &= -\frac{k}{\|\boldsymbol{B}_B\|^2}\dot{\boldsymbol{B}}_B \end{aligned} \tag{4.61}$$

Then the moment equation can be expressed as Equation 4.62.

$$\boldsymbol{L}_B = -\frac{k}{\|\boldsymbol{B}_B\|^2}[\dot{\boldsymbol{B}}_B\times]\boldsymbol{B}_B \tag{4.62}$$

A suitable method for determining the positive scalar gain k is provided in reference [7].

$$k = \frac{4\pi}{T_{orb}}(1 + sin\zeta_m)J_{min} \tag{4.63}$$

where

T_{orb} = Orbital period
i = Orbit inclination angle
ζ_m = Inclination of spacecraft orbit relative to the geomagnetic equatorial plane,
= given by $\zeta_m = \gamma_m + i$
J_{min} = Minimum principal moment of inertia
γ_m = Declination angle, angle between true north and magnetic north, 11.44 [deg]

In Figure 4.2, the reason for $\zeta_m = \gamma_m + i$ is illustrated, along with the angles.

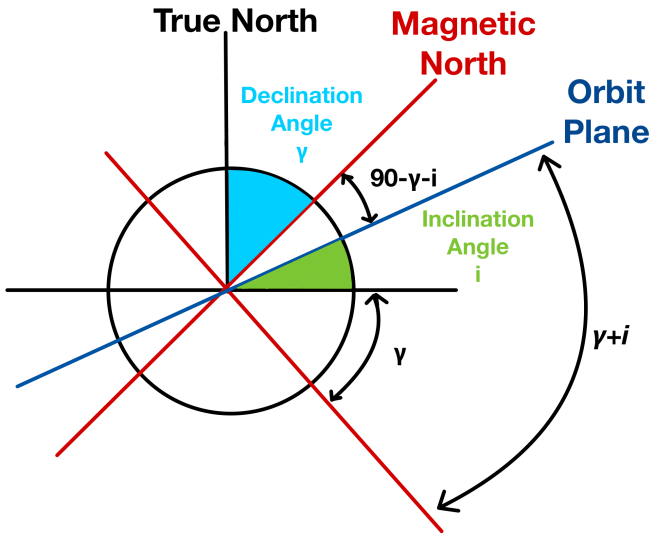


Figure 4.2: Illustration of the angles γ_m, i, and ζ_m

The B-dot controller is only Lyapunov stable, and not asymptotically Lyapunov stable, as proven in [40]. This is because the angular velocity and magnetic field vector may

not always be perpendicular and can even be parallel, which prevents the generation of a moment. This issue is especially noticeable when the spacecraft's orbit plane aligns with the geomagnetic equatorial plane. Another challenge arises near the magnetic poles, where the Earth's magnetic field is highly concentrated and closely spaced. As a result, the control law is not effective in these regions due to the limited changes observed in $\dot{\boldsymbol{B}}$.

4.2.2 Sliding-Mode Controller

The sliding-mode controller is a robust nonlinear controller often chosen when the precise mathematical model is either unknown or inaccurately modeled. For spacecraft attitude tracking, this controller proves advantageous due to uncertain factors like imprecisely known inertia matrices or the incomplete modeling of the space environment. Consequently, in this thesis, the sliding-mode controller is used as the primary attitude controller. This chapter is structured in two main sections. Initially, the theory of the sliding-mode controller is given, followed by its application to the spacecraft attitude tracking problem. The theoretical part is divided into six subparts. The first section defines the problem, addressing the challenges posed by the unknown mathematical model. The second part introduces an intermediate variable definition aimed at representing the dynamics as a first-order system. The third part offers a geometric interpretation of $s = 0$, where this hyperplane signifies both a spatial region and a dynamic property due to its invariance. Subsequently, the fourth section explores scenarios where s is not exactly 0, revealing its impact on the tracking error. The fifth part gives the sliding condition and the control law. Lastly, the issue of chattering is addressed, presenting a new control law. In the application part, the control law is formulated through the utilization of Euler's rotation formula and the attitude kinematics equation.

4.2.2.1 Problem Definition

In the Eq. 4.64, an n^{th} order nonlinear differential equation is given. In this system, f and b are not known exactly. If these are known exactly, u can be designed directly.

In general, the estimated $\hat{f}$ and exact f difference can be shown as in the Eq. 4.65. The same thing is true for the b. Also, the sliding-mode controller is most of the time used for trajectory tracking. In the Eq. 4.66, $\tilde{\boldsymbol{x}}$ is used to show the trajectory tracking error, which is the difference between the current state and the desired state.

$$\boldsymbol{x}^{(n)} = f(\boldsymbol{x}, \dot{\boldsymbol{x}}, \ddot{\boldsymbol{x}}, \dots, \boldsymbol{x}^{(n-1)}, t) + b(\boldsymbol{x}, \dot{\boldsymbol{x}}, \ddot{\boldsymbol{x}}, \dots, \boldsymbol{x}^{(n-1)}, t)\boldsymbol{u} \tag{4.64}$$

$$| \hat{f}(\boldsymbol{x}, t) - f(\boldsymbol{x}, t) | \leq F(\boldsymbol{x}, t) \tag{4.65}$$

$$\tilde{\boldsymbol{x}}(t) = \boldsymbol{x}(t) - \boldsymbol{x}_d(t) \tag{4.66}$$

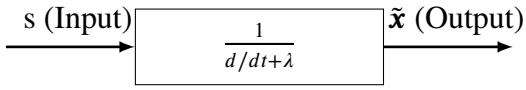
$\tilde{\boldsymbol{x}}(t)$ = Trajectory tracking error, $\boldsymbol{x}(t) - \boldsymbol{x}_d(t)$
$\boldsymbol{x}(t)$ = Current state
$\boldsymbol{x}_d(t)$ = Desired state

4.2.2.2 Intermediate Variable Definition and Purpose

The main objective in this part is to convert the n^{th} order problem into a first order problem. The reason for doing this is to make providing feedback for the desired trajectory more intuitive. In the first order problem, the feedback can be more easily expressed, for example, if a glass is placed on a desk at a specific position, and someone moves it to the left of that position, one can simply say to move it a little bit to the right [64]. However, expressing the same feedback for the n^{th} order problem is not as straightforward and clear. Therefore, to achieve the conversion of the system into a first order system, an intermediate variable is defined as s. This variable is not related to the Laplace variable used in control theory. The properties of this intermediate variable can be listed as follows:

1. $\dot{s}$ contains the control input u.
2. $s \to 0 \Rightarrow \tilde{\boldsymbol{x}} \to 0$.

To satisfy the first property, s can be defined as $s = \dot{\tilde{\boldsymbol{x}}}$ for the second order system. However, this definition does not fulfill the second property. On the other hand, to satisfy the second property, s can be defined as $s = \tilde{\boldsymbol{x}}$, but now the first property is not satisfied. Therefore, to meet both the first and second properties, s can be defined as $s = \dot{\tilde{\boldsymbol{x}}} + \lambda\tilde{\boldsymbol{x}}$. By defining $s = \dot{\tilde{\boldsymbol{x}}} + \lambda\tilde{\boldsymbol{x}}$, the first property is satisfied. For the validity of the second condition, $s = \dot{\tilde{\boldsymbol{x}}} + \lambda\tilde{\boldsymbol{x}}$ can be represented as a low-pass filter:



If a stable filter is used, then as $s \to 0$, the variable $\tilde{\boldsymbol{x}}$ also tends to approach zero. If generalization is done for the n^{th} order system, s can be defined as in the Eq. 4.67.

$$s = (\frac{d}{dt} + \lambda)^{n-1}\tilde{\boldsymbol{x}} \tag{4.67}$$

4.2.2.3 Geometric Interpretation of s=0

For the second order system, s can be expressed as $s = \dot{\tilde{\boldsymbol{x}}} + \lambda\tilde{\boldsymbol{x}}$. When plotting the phase portrait, Fig. 4.3 can be obtained for the case $s = 0$. Here, equating $s = 0$ leads to $\dot{\boldsymbol{x}}_d(t) + \lambda\boldsymbol{x}_d(t) = \dot{\boldsymbol{x}}(t) + \lambda\boldsymbol{x}(t)$. This implies that both $\dot{\boldsymbol{x}}_d(t)$ and $\boldsymbol{x}_d(t)$ lie on the $s = 0$ line, and the slope of the curve is $-\lambda$. Additionally, for the second order system, $s = 0$ represents a line, while for the n^{th} order system, $s = 0$ becomes a hyperplane and dividing the space into two regions: $s > 0$ and $s < 0$. Moreover, once $s = 0$ is satisfied, it remains invariant. This means that $s = 0$ is continuously satisfied, and the state-space trajectories slide along the motion. For a deeper understanding of the invariant property of $s = 0$, the reference [64, 65] is suggested. Overall, $s = 0$ represents both the system dynamics and its location in the phase space.

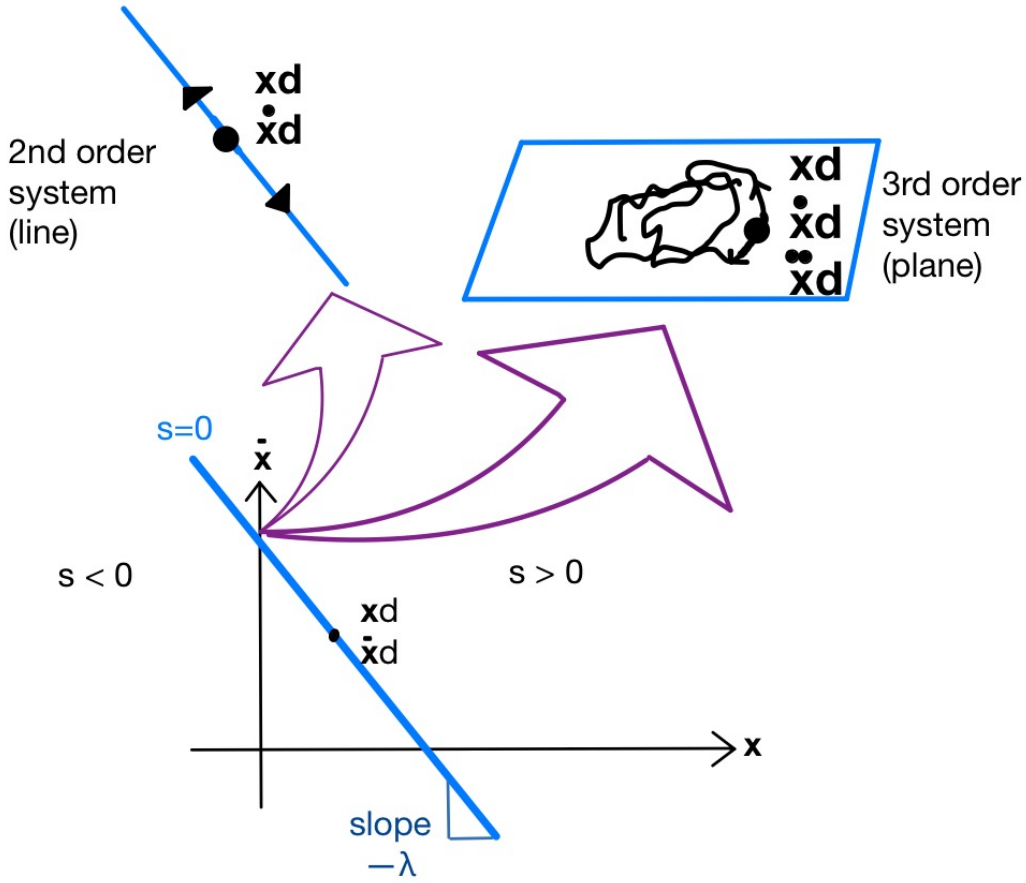


Figure 4.3: Phase diagram illustrating the behavior of a second-order system (In the second order system, state-space trajectories move up and down along a line, in the third order system, the state-space trajectories lie on a plane and can move freely within it.)

4.2.2.4 Error Analysis

As time approaches infinity and $\mid s \mid < \phi$ remains below a constant value ϕ, this error in ϕ influences and translates to the error in $\tilde{\boldsymbol{x}}$. In the case of the second-order system, where $s = \dot{\tilde{\boldsymbol{x}}} + \lambda\tilde{\boldsymbol{x}}$, if $\mid s \mid < \phi$, it follows that $\mid \tilde{\boldsymbol{x}} \mid < \phi/\lambda$. The detailed calculations can be found in the reference [64, 65]. For the n^{th} order system, there are $n-1$ first-order

low-pass filters, resulting in $\mid \tilde{\boldsymbol{x}} \mid < \phi/\lambda^{n-1}$.

4.2.2.5 Sliding Condition and Control Law

Firstly, the sliding condition is explained, and then the control law is derived based on the second-order system.

As previously emphasized, when the system is on the $s = 0$ hyperplane, it remains in that state. However, if the system is not initially on the hyperplane, reaching the hyperplane must be accomplished within finite time. The sliding condition guarantees this behavior. As illustrated in the Fig. 4.4 , if the initial dynamics of s are at point A, the system, aided by the sliding condition that ensures $s = 0$, must reach point B. Once the system reaches point B, it will remain there due to the invariance of $s = 0$.

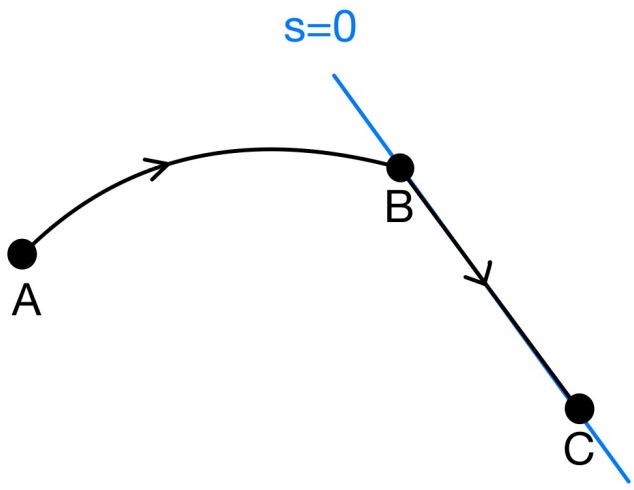


Figure 4.4: s dynamics both out of the hyperplane and on the hyperplane

The sliding condition can be expressed as shown in Equation 4.68.

$$\frac{1}{2}\frac{d}{dt}s^2 \leq -\eta \mid s \mid \tag{4.68}$$

where

η = Positive constant

Eq. 4.68 can be interpreted as follows: when s is not equal to 0, both terms involving s in the inequalities removing ($\dot{s} \leq -\eta$). This means that $\dot{s}$ is less than or equal to $-\eta$, indicating that s is a decreasing function.

In the explanation of the control law, a second-order nonlinear system is utilized, represented by $\ddot{\boldsymbol{x}} = f + u$. This results in $\dot{s} = f + u - \ddot{\boldsymbol{x}}_d + \lambda\dot{\tilde{\boldsymbol{x}}}$. However, when the exact mathematical model is not known or not used, the expression becomes $\dot{s} = \hat{f} + \hat{u} - \ddot{\boldsymbol{x}}_d + \lambda\dot{\tilde{\boldsymbol{x}}}$ due to the unavailability of the exact model. Since $s = 0$ is an invariant condition, this implies $\dot{s} = 0$, which leads to $\hat{u} = -\hat{f} + \ddot{\boldsymbol{x}}_d - \lambda\dot{\tilde{\boldsymbol{x}}}$. If the exact mathematical model is used with the exact u, then Eq. 4.69 can be written. In fact, when $\hat{u}$ and $\hat{f}$ are used, $\dot{s} = 0$ is satisfied, as $\hat{u}$ is chosen according to this, leading to Eq. 4.70.

$$\dot{s} = f + u - \ddot{\boldsymbol{x}}_d + \lambda\dot{\tilde{\boldsymbol{x}}} \tag{4.69}$$

$$\begin{aligned} \dot{s} &= \hat{f} + \hat{u} - \ddot{\boldsymbol{x}}_d + \lambda\dot{\tilde{\boldsymbol{x}}} \\ 0 &= \hat{f} + \hat{u} - \ddot{\boldsymbol{x}}_d + \lambda\dot{\tilde{\boldsymbol{x}}} \end{aligned} \tag{4.70}$$

Also, from the feedback law $u = \hat{u} - ksgn(s)$, we can understand that it is the main reason for reducing the system to first-order dynamics. By setting $s = 0$, the sign of s is inverted and extract from $\hat{u}$ with a coefficient. Therefore, with this information and Eqs. 4.69 and 4.70, $\dot{s}$ can be expressed as shown in Eq. 4.71.

$$\begin{aligned} \dot{s} &= f - \hat{f} + u - \hat{u} \\ &= f - \hat{f} - ksgn(s) \end{aligned} \tag{4.71}$$

If Eq. 4.71 is used in Eq. 4.68, which is the sliding-condition equation, Eq. 4.72 can be expressed.

$$\frac{1}{2}\frac{d}{dt}s^2 = s\dot{s} = s(f - \hat{f}) - k \mid s \mid \leq -\eta \mid s \mid \tag{4.72}$$

Eqs. 4.65 and 4.72 imply that the parameter k can be chosen as $k = F + \eta$.

4.2.2.6 Chattering

Chattering is another significant aspect of the sliding-mode controller. Due to the discontinuous control input u and the presence of measurement or actuator lags, exact satisfaction of $s = 0$ is not always achieved. Instead, as illustrated in Fig. 4.5, the system operates slightly above and below $s = 0$. This behavior is often undesirable as it can lead to the excitation of high-frequency dynamics in the spacecraft.

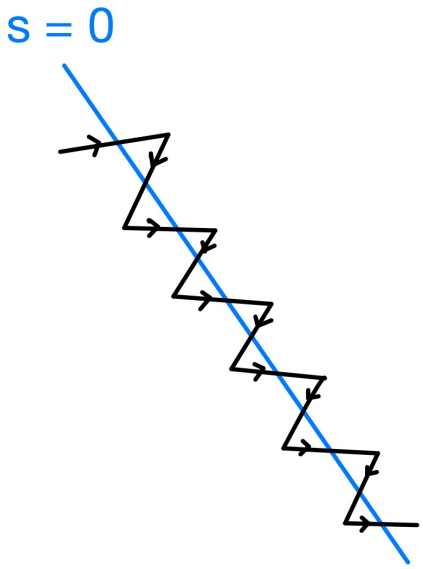


Figure 4.5: Chattering

One solution to address this problem is to utilize a continuous control law. Instead of employing the $sgn(s)$ function, a saturation function $sat(s)$ can be utilized. Both functions are depicted in Fig. 4.6. The saturation function remains continuous within the range of ± 1. However, in practice, the desire to adjust this range of ± 1 arises.

Consequently, a modified saturation function is employed to ensure continuity of the input within the range of $\pm\phi$.

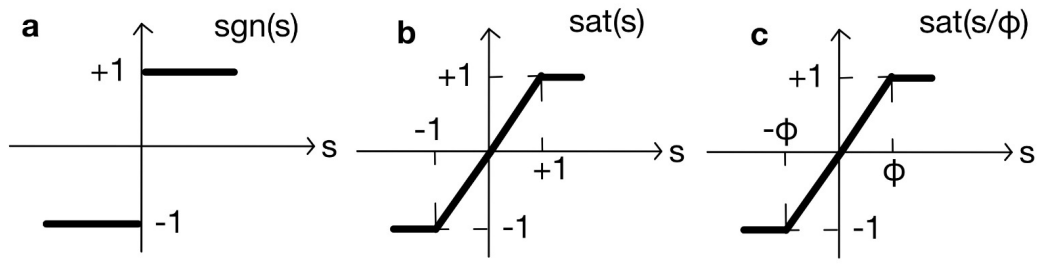


Figure 4.6: **a)** sign function, **b)** saturation function which is used in the control theory, **c)** modified saturation function

The values of $sgn(s)$ and $sat(s/\phi)$ are equivalent when s lies outside the range of ϕ and $-\phi$ ($s > \phi$ and $s < -\phi$). As a result, all the principles and concepts explained about the sliding-mode controller remain valid within this specific region. In other words, the invariance of $s = 0$ now holds for s within the interval $[-\phi, \phi]$. Once the state of the system is within this interval (s between ϕ and $-\phi$), it will stay in that range, similar to a boundary layer in fluid dynamics, as shown in Fig. 4.7.

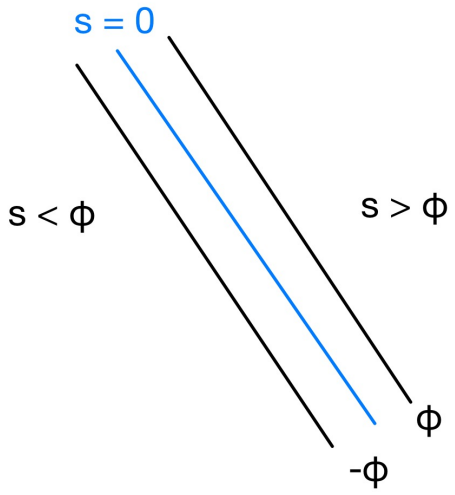


Figure 4.7: Geometric interpretation of utilizing $sat(s/\phi)$ as a boundary layer

4.2.2.7 Sliding-mode Controller Application on the Spacecraft Attitude Determination

When implementing the sliding-mode controller for spacecraft attitude tracking, the primary reference utilized is [40].

The intermediate variable s is defined as presented in Eq. 4.73.

$$s = (\boldsymbol{\omega} - \boldsymbol{\omega}_d) + \lambda \delta \boldsymbol{q}_{1:3} \tag{4.73}$$

The quantity $\delta \boldsymbol{q}_{1:3}$ can be interpreted as the derivative of $\tilde{\boldsymbol{\omega}}$, where this attitude kinematics is influenced by the angular velocity. This relationship is evident from Eq. 3.24.

$\tilde{\boldsymbol{\omega}}$ can be seen as the derivative of the $\delta \boldsymbol{q}_{1:3}$. Reason behind this attitude kinematics is depend on the angular velocity this can be seen clearly from the Eq. 3.24. However, because there are two quaternions representing the same attitude and one causes the spacecraft to rotate more longer than the other, this situation is undesirable. To address this, the following control law ensures that the spacecraft reorients itself along the shortest path, as shown in Equation 4.74:

$$s = (\boldsymbol{\omega} - \boldsymbol{\omega}_d) + \lambda sgn(\delta \boldsymbol{q}_4) \delta \boldsymbol{q}_{1:3} \tag{4.74}$$

To derive the suitable control law, we can set $\dot{s} = 0$ by equating it to zero. As a result, $\dot{s}$ can be defined as follows:

$$\dot{s} = (\dot{\boldsymbol{\omega}} - \dot{\boldsymbol{\omega}}_d) + \lambda sgn(\delta \boldsymbol{q}_4) \delta \dot{\boldsymbol{q}}_{1:3} \tag{4.75}$$

In Equation 4.75, Equation 3.27 or Equation 3.28 can be substituted, depending on whether a reaction wheel is utilized or not, for $\dot{\boldsymbol{\omega}}$. Moreover, Equation 4.36 can be used instead of $\dot{\boldsymbol{q}}_{1:3}$, and by replacing $\hat{\boldsymbol{\omega}}$ with $\boldsymbol{\omega}_d$, $\dot{s}$ can be made equal to zero, and $\hat{\boldsymbol{L}}$, which represents $\hat{\boldsymbol{u}}$, can be calculated.

$$\hat{\boldsymbol{L}} = J\left\{\frac{\lambda}{2}\left[\mid \delta\boldsymbol{q}_4 \mid (\boldsymbol{\omega}_d - \boldsymbol{\omega}) - sgn(\delta\boldsymbol{q}_4)\delta\boldsymbol{q}_{1:3} \times (\boldsymbol{\omega}_d + \boldsymbol{\omega})\right] + \dot{\boldsymbol{\omega}}_d\right\} + [\boldsymbol{\omega}\times]J\boldsymbol{\omega} \quad (4.76)$$

The feedback control law is expressed as shown in Eq. 4.77.

$$\boldsymbol{L} = \hat{\boldsymbol{L}} - ksat(s/\phi) \quad (4.77)$$

where

k = JG
G = Positive constant

Furthermore, the modified saturation function can be explicitly expressed:

$$sat(s/\phi) = \begin{cases} 1, & s > \phi \\ s/\phi, & \mid s \mid < \phi \\ -1, & s < -\phi \end{cases} \quad (4.78)$$

Choosing the values of ϕ, k, and λ depends on the specific problem. These parameters are adjusted to match the problem's requirements. One can find the parameter values used for this application in Table 7.1.

4.2.3 Reaction Wheel Desaturation Controller

Reaction wheel desaturation control is utilized for the momentum damping of the reaction wheel. In this control, magnetorquer is used again to damp the reaction wheels. The same controller law as the B-dot control is used again. In Eq. 4.56, $\boldsymbol{\omega}$ is replaced by $\boldsymbol{h}$. This is demonstrated in Eq. 4.79. The moment equation for the reaction wheel desaturation control is again written using Eq. 4.57. The appropriate value of the k gain can be selected based on the specific characteristics of the spacecraft.

$$\boldsymbol{m} = \frac{k}{\|\boldsymbol{B}\|}\boldsymbol{h} \times \boldsymbol{b} \quad (4.79)$$

where

$\boldsymbol{h}$ = Angular momentum of the reaction wheel

CHAPTER 5

FAULT DETECTION AND ISOLATION ALGORITHM

This chapter is dedicated to explaining the proposed FDI methodology, which is essential for spacecraft due to the numerous risks involved. The FDI structure proposed in this thesis is based on BN, as introduced in Section 5.1, providing the necessary foundation for understanding the FDI methodology.

The advantages of utilizing BN in the FDI algorithm are discussed in Section 5.2.

In Section 5.3, the construction of the BN is described, involving the division of the MEKF fault/error source into four parts: QUEST, gyroscope, star tracker, and mathematical nodes. In Section 5.4, a detailed examination is carried out for each of these parts to identify the nodes that can be observed within them.

These four structures are interconnected through switches that are activated based on the filter type, whether it's MEKF with gyroscope measurement or MEKF with mathematical model. The activation of switches also depends on the source of quaternions. This process is detailed in Section 5.5.

To provide a deeper understanding of the algorithm's logic, a working example of the algorithm is presented at the end of the chapter.

5.1 Bayesian Networks

Inference is the process of obtaining information or insight from data or observations. According to the reference [60], there are two main processes in the inference aspect:

1. Calculation of the posterior probability, which can be expressed as $p(\boldsymbol{x} \mid \boldsymbol{y})$

where $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$.

2. Calculation of the maximum a posteriori (MAP) estimate, which is denoted as $\hat{\boldsymbol{x}} \in \arg\max_{\boldsymbol{x}} p(\boldsymbol{x} \mid \boldsymbol{y})$.

The first aspect can be thought of as finding the probability of $\boldsymbol{x}$ when $\boldsymbol{y}$ is known, while the second involves finding the $\boldsymbol{x}$ that maximizes $p(\boldsymbol{x} \mid \boldsymbol{y})$. It can be challenging to solve these problems, however Probabilistic Graphical Models (PGMs) have the advantage of decreasing the cost of the calculation and simplifying the inference process with graphs.

BNs are a type of PGM that are mathematically defined as Directional Acyclic Graphs (DAGs). In PGMs, nodes represent random variables and edges between nodes denote the relationships between them. These relationships in BNs are directional, indicating that there is a particular direction for information to flow between nodes. This direction is represented by arrows in the graph. Additionally, BNs are acyclic, meaning that the arrows do not form cycles. For further details on BNs, following references can be used [12, 18, 34, 38, 44].

BNs can represent any joint probability distribution between random variables. In general, the joint probability distribution of n random variables can be expressed as:

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n) = p(\boldsymbol{x}_1)p(\boldsymbol{x}_2 \mid \boldsymbol{x}_1) \dots p(\boldsymbol{x}_n \mid \boldsymbol{x}_1, \dots, \boldsymbol{x}_{n-1}) \tag{5.1}$$

The right-hand side of Equation 5.1 can be simplified using BNs. Specifically, conditional probabilities can be represented by parent and child nodes.

As an example, Figure 5.1 illustrates a BN. The joint probability distribution of random variables $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$ can be expressed in general as:

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) = p(\boldsymbol{x}_1)p(\boldsymbol{x}_2 \mid \boldsymbol{x}_1)p(\boldsymbol{x}_3 \mid \boldsymbol{x}_1, \boldsymbol{x}_2) \tag{5.2}$$

However, if we use the properties of this joint distribution as a BN shown in Fig.5.1, one can simplify the expression to:

$$p(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) = p(\boldsymbol{x}_1)p(\boldsymbol{x}_2 \mid \boldsymbol{x}_1)p(\boldsymbol{x}_3 \mid \boldsymbol{x}_1) \tag{5.3}$$

This example shows how BNs and other PGMs can be described and factorized to simplify distributions and calculation.

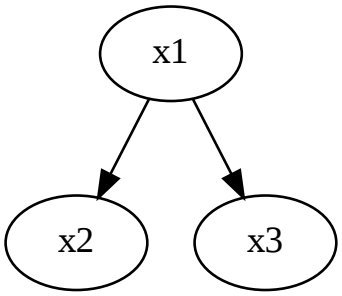


Figure 5.1: An illustration of a Bayesian Network with random variables $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$.

5.2 Bayesian Network Usage in the FDI Algorithm

As mentioned previous section, random variable relations can be represented by the help of the graph, such as conditional independency and causal/evidential relationship.

There are some reasons behind using the BN in the FDI algorithms. These can be itemized:

- Clarifying the relationship between faults/errors and their probabilities
- Using declarative representation, which means separation of the knowledge and reasoning (or model) [34] facilitates the re-usage of the algorithms for new missions and spacecraft
- Understanding fault/error reasons more deeply

A BN is constructed to help the FDI process. The BN is built by considering the MEKF fault/error sources in detail. Figure 5.2 shows the BN, which consists of four main parts: QUEST faults, gyroscope faults, star tracker faults, and mathematical model mismatch. The following subsections provide further details about these four groups of nodes. The observable nodes in the BN and their observation principles are also explained. Also, in Section 5.5, one can find a detailed explanation of the switches shown in Figure 5.2.

The conditional probability distributions utilized in the BN are determined using expert knowledge, which is given as an tables in the Appendix section. All random variables are assumed to be binary, where "1" denotes the occurrence of a fault/error and "0" denotes the absence of a fault/error. The probabilities are computed using exact inference via variable elimination, with the aid of the Python pgmpy library [5]. Variable elimination approach is not explained in this thesis, however readers can get more information at [34].

5.3 Node Determination

This section provides a detailed analysis of faults and errors, which are categorized into four parts: QUEST, gyroscope, star tracker, and mathematical model. These faults and errors are studied as random variables, and a Bayesian network is constructed accordingly.

5.3.1 Fault/Error Analysis for QUEST

This subsection explains the analysis of four primary faults/errors in the QUEST system. The faults in the sensor and reference model for both the sun direction vector and magnetic field vector are investigated.

1. Sun Sensor Faults: This section deals with the possible sun sensor faults. The sensor may not be able to measure anything if an eclipse occurs, or if there is an albedo effect on the sensor. Calibration of the sensor may also present problems.

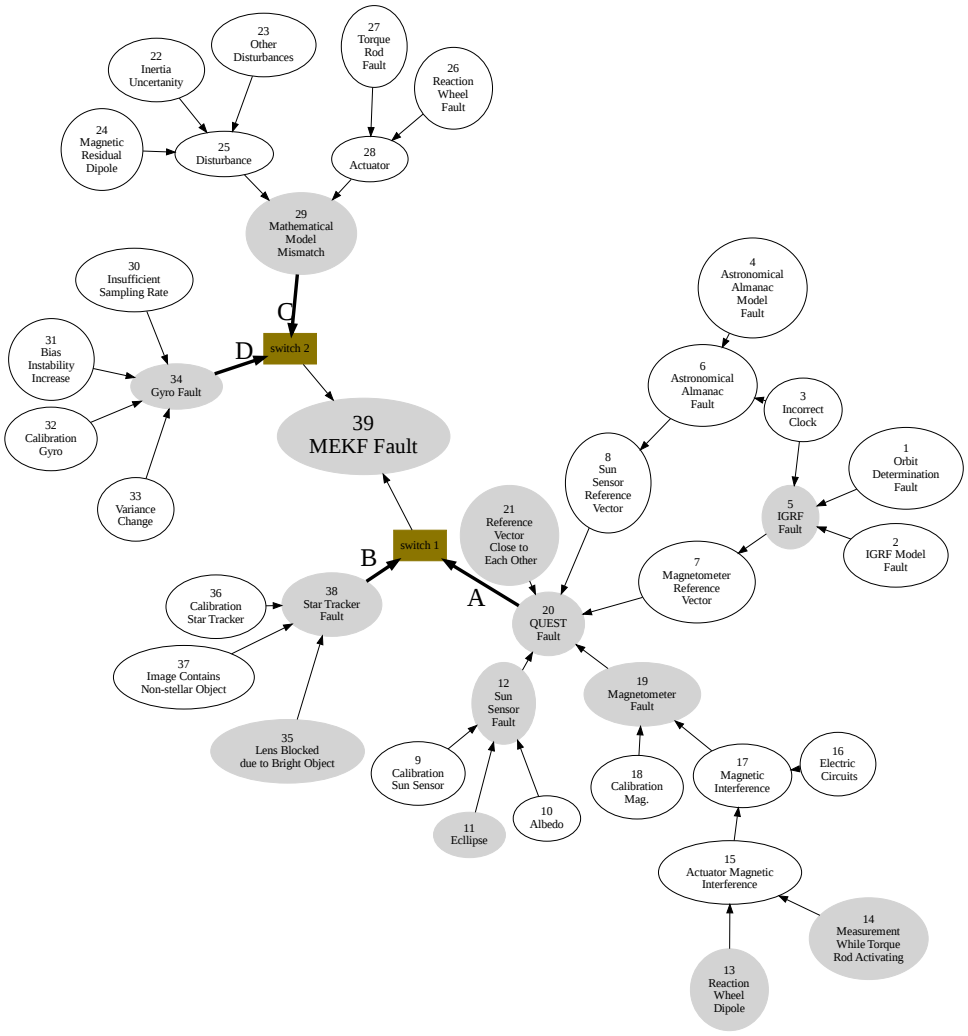


Figure 5.2: BN for MEKF FDI, observable nodes are shown as shaded

2. Magnetometer Faults: This section deals with magnetic interference that can cause problems in the system. Such interference may be caused by electric circuits, as well as actuator magnetic interference (e.g., from reaction wheel and magnetorquer). The calibration of the sensor may also cause issues.

3. Sun Sensor Reference Vector: Problems may occur due to errors in the Astronomical Almanac. This can happen for two reasons: the model may be incorrect, or the clock.

4. Magnetometer Reference Vector: Errors may arise due to inaccuracies in the International Geomagnetic Reference Field (IGRF), or due to faults/errors in the data provided to IGRF (e.g., orbit determination errors or an incorrect clock).

5.3.2 Faults/Errors Analysis for Gyroscope

The following are the faults associated with the gyroscope:

1. Insufficient Sampling Rate: The sampling rate of the gyroscope may be inadequate. The filtered result of the gyroscope may be used, and the filter embedded in the gyroscope may have an insufficient sampling rate for the motion.

2. Bias Instability Increase: Accurate determination of the gyroscope bias is crucial for the use of gyroscope measurements in the MEKF. Thus, inaccuracies in the gyroscope bias must be accounted for in FDIR algorithms.

3. Non-calibrated Gyroscope: The use of a non-calibrated gyroscope may result in incorrect data.

4. Variance Change: The angular random walk of the gyroscope is typically provided in the data sheets of the gyro. However, this value may also change due to factors such as temperature. Therefore, it is also included in the list of faults.

5.3.3 Faults/Errors Analysis for Star Tracker

Although star trackers offer higher accuracy than other sensors utilized in this thesis, they also have unique errors that must be considered. These errors include:

1. Lens Blockage due to Bright Objects: The lens of the star tracker may become blocked by bright objects such as the Sun.

2. Star Tracker Calibration: Problems may arise during the calibration of the star tracker.

3. Non-Stellar Objects in Image: The image captured by the star tracker may contain non-stellar objects, such as satellites, which are not included in the sensor's catalogs. This can lead to inaccuracies and potential issues.

5.3.4 Faults/Errors Analysis for Mathematical Model Mismatch

The mathematical model used in this thesis is based on the Euler equations. Torque is calculated as the sum of active sensor torques using Equations 3.26 and 3.27. Hence, the mismatches in the mathematical model can be divided into two parts:

1. Actuator Fault: This refers to the uncertainty in the torque values of magnetorquers and reaction wheels due to their internal properties.

2. Disturbances: These are external factors that affect the spacecraft's attitude, such as inertia uncertainty and magnetic dipole moment.

5.4 Observable Nodes

Observation refers to the process of obtaining actual values for certain nodes, which in turn provides valuable information about other nodes in the BN. This can be represented as a posterior probability distribution. In the BN, some nodes are observable, and they can be classified into four distinct categories: Observable nodes at QUEST, gyroscope, star tracker, and mathematical model. Each of these categories is examined separately.

5.4.1 Observable Nodes at QUEST

- IGRF Fault Node: To gain insight into the IGRF error, one can compare the magnitude of the magnetometer vector and the IGRF vector. Despite being in different resolution frames, the magnitudes of these vectors should be equal.
- Eclipse Node: The occurrence of an eclipse can be observed for a spacecraft by monitoring its solar power. If the solar power of the spacecraft increases, it indicates that the satellite is no longer in an eclipse.
- Magnetometer Fault and Sun Sensor Fault Nodes: Typically, a spacecraft is equipped with multiple magnetometers and sun sensors. To detect faults and errors, a binary combination of all sensors can be performed, and the resulting difference between sensor readings can be considered as a fault or error. Additionally, a comparison can be made between the values obtained and the $\pm 6\sigma$ range, which can be obtained from sensor catalogs.
- Measurement While Torque Rod Activating and Reaction While Dipole Nodes: The magnetometer measures the magnetic field, but the presence of a torque rod can generate an additional magnetic field, leading to errors in magnetometer readings. Similarly, when the reaction wheel is active, it can also create a magnetic field that affects the magnetometer measurements. These nodes can be considered observable since the active sensors are known and can be monitored.
- QUEST Fault Node: Both QUEST and star tracker provide quaternion measurements, which can be compared using an algorithm to determine the difference in attitude. If the difference exceeds a predefined threshold value, the system can send error messages. Additionally, the results from both QUEST and star tracker can be cross-checked against each other. This comparison can also be made using the quaternion results obtained from the MEKF.
- Reference Vectors Close to Each Other Node: If the IGRF and sun direction reference vector, or the magnetometer and sun sensor measurement vector directions are close, it can result in a QUEST fault or error. This is because the QUEST algorithm is unable to provide accurate attitude determination if the attitude determination vectors are parallel to each other. This can be checked using the dot product between the vectors to ensure that they are not parallel.

5.4.2 Observable Nodes at Gyroscope

- Gyro Fault Node: To observe the gyroscope fault, a threshold value can be set based on the expected maneuver and orbit of the spacecraft. If the gyro measurements exceed this predetermined threshold value, an error message can be sent to the system. Furthermore, the filter covariance matrix can be examined to identify any inconsistencies or irregularities in the gyro measurements.

5.4.3 Observable Nodes at Star Tracker

- Lens Blocked due to Bright Object Node: By determining the spacecraft's attitude and position, it can be determined whether the star tracker has the sun or the planet in its field of view.
- Star Tracker Fault Node: The technique used for detecting QUEST faults can also be applied to identify star tracker faults.

5.4.4 Observable Nodes at Mathematical Model

- Mathematical Model Mismatch Node: A comparison can be made between the angular velocities obtained from the gyroscope and the mathematical model. If there is a significant difference between the two, it indicates a fault or error in the mathematical model.

5.5 MEKF with Different Measurements

MEKF, like all filters, consists of both a mathematical model and a measurement stages. In order to provide the filter with quaternion measurements, there are two sources available: QUEST and star tracker. Additionally, since quaternion and angular velocity are kinematically related, the angular velocity must also be utilized for the time propagation of the quaternion. The angular velocity can be obtained by using the gyro measurements or by means of the Euler rotations. These components are depicted schematically in Fig. 5.3. As a result, four combinations are produced, each with a different graph structure. These combinations are illustrated in Fig. 5.4.

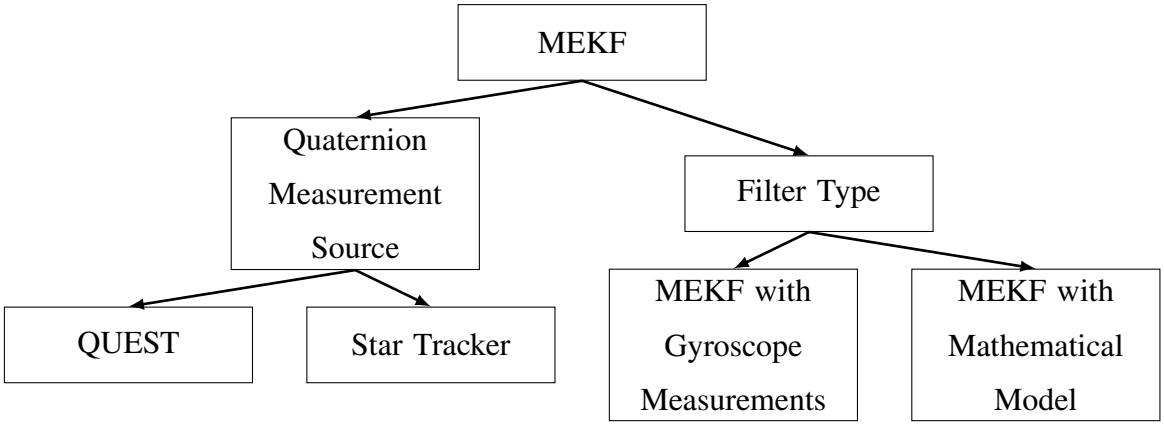


Figure 5.3: Schematic of the different quaternion measurement source and filter type

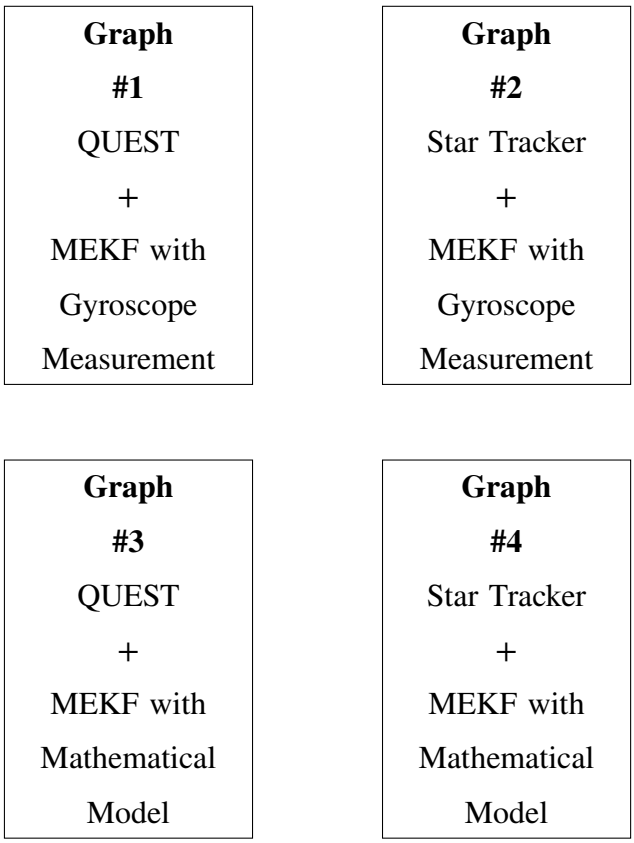


Figure 5.4: Four combination shows quaternion measurement source and filter type

To achieve using of different measurement sources and filter, two different switches are employed in the BN, as illustrated in Fig. 5.2. Additionally, Table 5.1 provides a summary of the switch structure and graph for each of the four different combina-

tions.

Table 5.1: Switch positions and active sides of the graph

Graph #	***Switch 1***		***Switch 2***		***Active Sides***
	A	**B**	**C**	**D**	
1	1	0	0	1	QUEST + MEKF with Gyro. Mea.
2	0	1	0	1	Star Tracker + MEKF with Gyro. Mea.
3	1	0	1	0	QUEST + MEKF with Math. Model
4	0	1	1	0	Star Tracker + MEKF with Math. Model
Note: 1 stands for the switch is on position					

Given that QUEST and star tracker serve as alternative measurement sources for the MEKF, in case of a fault or error in one of them, the other can be used as a backup. This applies to both the gyroscope measurement and the mathematical model. In order to understand the cause of the fault or error in the MEKF result, BN is utilized. To illustrate this logic more precisely, consider the following example:

Working Example of Algorithm

1. Error is observed at MEKF

2. Understand which combination is used (Graph #1 or Graph #2 or Graph #3 or Graph #4)

Assume: Graph #1 is used.

3. Observe all observable nodes in the Graph #1.

4. Run the Inference Algorithm

5. Find the Result of these queries and compare:

$$P(Quest\ Fault\ |MEKF\ Fault, Other\ Observable\ Nodes)$$
$$>$$
$$=$$
$$<$$
$$P(Gyro\ Fault\ |MEKF\ Fault, Other\ Observable\ Nodes)$$

6. If $>$ $\Rightarrow$ Replace Graph #1 with Graph #2
   If $<$ $\Rightarrow$ Replace Graph #1 with Graph #3
   If $=$ $\Rightarrow$ Replace Graph #1 with Graph #2 and Graph #3, then; recheck both of them.

CHAPTER 6

HYBRID ATTITUDE DETERMINATION CONTROL DESIGN

This chapter provides a comprehensive explanation of the ADCS modes, including their specific purposes and functionalities. The algorithms discussed in Chapter 4, such as the estimators and controllers, are employed within these modes, and their compositions are thoroughly described in this chapter. Additionally, mode transitions are described using both threshold values and Bellman optimization.

Section 6.1 provides explanations of all the modes in the ADCS. It also presents the composition algorithms used to combine the estimator and controllers for each mode.

In Section 6.2, mode transitions are performed using threshold values. The construction of these thresholds is explained in detail.

Section 6.3 presents an alternative method for mode transition. The spacecraft ADCS is modeled as a discrete state Markov chain, and Bellman optimization is employed to facilitate the mode transitions.

6.1 Modes of the ADCS

Each mode is explained in detail, outlining its functionalities.

6.1.1 Mode 1

Mode 1 is employed to initiate the mode transition process.

6.1.2 Mode 2

During the separation phase of the spacecraft, a controller known as B-dot is employed for control. This controller is especially beneficial as the spacecraft experiences rapid rotation during this stage, and B-dot helps to reduce the angular velocity. Additionally, the QUEST algorithm is utilized to estimate the coarse orientation of the spacecraft. However, it is important to mention that the MEKF algorithm is not suitable for this phase since the spacecraft's rotation is still too fast for it to work effectively. The algorithm is presented in Algorithm 3 for this mode.

Algorithm 3 MODE 2 (QUEST + B-dot)

Inputs : σ_{mag}: standard deviation of the magnetometer measurement noise

σ_{sun}: standard deviation of the sun sensor measurement noise

$\boldsymbol{m}_{B,k}$: magnetometer measurement at time k

$\boldsymbol{m}_{B,k+1}$: magnetometer measurement at time k+1

$\boldsymbol{s}_{B,k}$: sun sensor measurement at time k

$\boldsymbol{m}_{R,k}$: magnetic field at ECI at time k

$\boldsymbol{s}_{R,k}$: sun direction at ECI at time k

$\boldsymbol{J}$: Moment of inertia

dt: time incrementation

Outputs: $\hat{\boldsymbol{q}}_k$: estimated quaternion at time k

$\boldsymbol{L}$: commanded moment to magnetorquer

$w \leftarrow [1/\sigma_{mag}, 1/\sigma_{sun}]$

```
// The measurement weights utilized in QUEST are
   inversely proportional to the measurement standard
   deviation.
```

$w \leftarrow w/norm(w)$

```
// It is necessary for the summation of the weights to
   equal 1.
```

$b \leftarrow [\boldsymbol{m}_{B,k}/\|\boldsymbol{m}_{B,k}\| \;\; \boldsymbol{s}_{B,k}/\|\boldsymbol{s}_{B,k}\|]$

$r \leftarrow [\boldsymbol{m}_{R,k}/\|\boldsymbol{m}_{R,k}\| \;\; \boldsymbol{s}_{R,k}/\|\boldsymbol{s}_{R,k}\|]$

```
// In the QUEST method, the vectors used must be unit
   vectors.
```

$\hat{\boldsymbol{q}}_k \leftarrow Quest(w, r, b)$

$L \leftarrow B\_dot(\boldsymbol{J}, \boldsymbol{m}_{R,k}, \boldsymbol{m}_{R,k+1}, dt)$

6.1.3 Mode 3

In this mode, once the spacecraft velocity has decreased, the MEKF dynamic attitude determination algorithm is employed. However, as discussed in Section 5.5, the MEKF can use at four different combinations. In this particular mode, the star tracker quaternion measurements are not utilized. This decision was made due to the fact that

the spacecraft's velocity may still be too high for the star tracker to function optimally. Since this sensor relies on the camera, if the spacecraft is rotating too quickly, the star tracker may not perform star identification successfully. As a result, only the QUEST quaternion measurement is used in this mode, as used in the Filter or Graph #1 or #3 shown in Figure 5.4. To control the spacecraft's orientation during this mode, a B-dot algorithm is employed. This decision was made because the filter's convergence or effectiveness during this phase is uncertain. Therefore, a control algorithm that does not depend on the filter's estimations, such as B-dot, is used. Reason behind is that the sliding-mode controller relies on the filter's estimations. Algorithm 4 presents the more information about the algorithm.

Algorithm 4 MODE 3 (MEKF + B-dot)

Inputs : σ_{mag}: standard deviation of the magnetometer measurement noise

σ_{sun}: standard deviation of the sun sensor measurement noise

$\boldsymbol{m}_{B,k}$: magnetometer measurement at time k

$\boldsymbol{m}_{B,k+1}$: magnetometer measurement at time k+1

$\boldsymbol{s}_{B,k}$: sun sensor measurement at time k

$\boldsymbol{m}_{R,k}$: magnetic field at ECI at time k

$\boldsymbol{s}_{R,k}$: sun direction at ECI at time k

$\boldsymbol{J}$: Moment of inertia

dt: time incrementation

$which\_filter$: 1 = MEKF with gyroscope measurements or 2 = MEKF with mathematical model

$\boldsymbol{x}_{k|k-1}$: Global states, differs according the $which\_filter$

$\Sigma_{k|k-1}$: Covariance matrix, differs according the $which\_filter$

$\tilde{\boldsymbol{\omega}}_k$: Gyroscope measurement at time k, which is used if the $which\_filter = 1$

Outputs: $\hat{\boldsymbol{x}}_{k|k}$: estimated global states at time k

$\hat{\boldsymbol{x}}_{k+1|k}$: estimated global states at time k+1

$\Sigma_{k|k}$: estimated covariance at time k

$\Sigma_{k+1|k}$: estimated covariance at time k+1

$\boldsymbol{L}$: commanded moment to magnetorquer

$w \leftarrow [1/\sigma_{mag}, 1/\sigma_{sun}]$

$w \leftarrow w/norm(w)$

$b \leftarrow [\boldsymbol{m}_{B,k}/\|\boldsymbol{m}_{B,k}\| \;\; \boldsymbol{s}_{B,k}/\|\boldsymbol{s}_{B,k}\|]$

$r \leftarrow [\boldsymbol{m}_{R,k}/\|\boldsymbol{m}_{R,k}\| \;\; \boldsymbol{s}_{R,k}/\|\boldsymbol{s}_{R,k}\|]$

$\tilde{\boldsymbol{q}}_k \leftarrow Quest(w, r, b)$

// The QUEST estimation is utilized as a measurement input to the filter.

$L \leftarrow B\_dot(\boldsymbol{J}, \boldsymbol{m}_{R,k}, \boldsymbol{m}_{R,k+1}, dt)$

If $which\_filter = 1$ **then**

$[\hat{\boldsymbol{x}}_{k|k}, \hat{\boldsymbol{x}}_{k+1|k}, \Sigma_{k+1|k}] \leftarrow MEKF\_1(\hat{\boldsymbol{x}}_{k|k-1}, \Sigma_{k|k-1}, \tilde{\boldsymbol{q}}_k, \tilde{\boldsymbol{\omega}}_k, dt)$

// $MEKF\_1$ = MEKF with gyro measurements

// The variable $\boldsymbol{\omega}_k$ serves as an input measurement to the $MEKF\_1$ algorithm.

else

$\boldsymbol{h} \leftarrow 0_{3\times3}$

$\dot{\boldsymbol{h}} \leftarrow 0_{3\times3}$

// In this mode, only magnetorquers are employed, and as a result, the reaction wheel parameters are set to zero.

$[\hat{\boldsymbol{x}}_{k|k}, \hat{\boldsymbol{x}}_{k+1|k}, \Sigma_{k+1|k}] \leftarrow MEKF\_2(\hat{\boldsymbol{x}}_{k|k-1}, \Sigma_{k|k-1}, \tilde{\boldsymbol{q}}_k, \boldsymbol{L}, \boldsymbol{h}, \dot{\boldsymbol{h}}, J, dt)$

// $MEKF\_2$ = MEKF with mathematical model

6.1.4 Mode 4

In this mode, the spacecraft uses both a sliding mode controller and a dynamic attitude determination algorithm called MEKF. The spacecraft's angular velocity is low during this mode, so all available measurement sources are used with two different filters to ensure accurate estimation. This allows for the use of a FDI algorithm like BN. If there is an error or fault in one of the sensors, BN can be activated, and the filters and measurement sources can be switched. This improves the spacecraft's overall mission performance because this mode is the primary mission mode of the hybrid attitude determination and control algorithms. Since the angular velocity is low, the controller can use the filter estimations instead of B-dot, so a sliding mode controller is used. To use the sliding-mode controller, the desired quaternions and angular velocities must be given. The details of how to obtain the desired quaternions and angular velocity are explained in the Algorithm 5. Algorithm 6 provides a more detailed explanation of the algorithm used in this mode.

Algorithm 5 Desired Angular Velocity for the General Case

Inputs : $\dot{\phi}, \dot{\theta}, \dot{\psi}$: The angular rates are calculated based on the desired motion

ϕ, θ, ψ: The angles between LVLH and body frame are calculated based on the desired motion

$\boldsymbol{r}_{I,k}$: position of the spacecraft resolved at the inertial frame at time k

$\boldsymbol{v}_{I,k}$: velocity of the spacecraft resolved at the inertial frame at time k

$\dot{\boldsymbol{v}}_{I,k}$: time change of the velocity of the spacecraft resolved at the inertial frame which is ECI at time k

Outputs: $\boldsymbol{q}_{c,k}$: desired quaternion at time k

$\boldsymbol{\omega}_{c,k}$: desired angular velocity of the spacecraft with respect to inertial frame, resolved at the body frame at time k

$A_{BO} \leftarrow A(\boldsymbol{e}_1, \psi)A(\boldsymbol{e}_2, \theta)A(\boldsymbol{e}_3, \phi)$

`// The transformation matrix from the LVLH frame to the body frame can be obtained using Eq.3.4.`

$A_{IO} \leftarrow \begin{bmatrix} \boldsymbol{o}_{1I} & \boldsymbol{o}_{2I} & \boldsymbol{o}_{3I} \end{bmatrix}$

$A_{OI} \leftarrow A_{IO}^T$

`// The transformation matrix from LVLH to inertial frame can be obtained using the Eq.3.1.`

$A_{BI} \leftarrow A_{BO}A_{OI}$

$\boldsymbol{q}_{c,k} \leftarrow quat2dcm(A_{BI})$

`// The desired quaternion is obtained.`

$\boldsymbol{\omega}_B^{BO} \leftarrow function(\dot{\phi}, \dot{\theta}, \dot{\psi}, \phi, \theta, \psi)$

`// The equation can be found in Eq.3.6.`

$\boldsymbol{\omega}_O^{OI} \leftarrow function(\boldsymbol{r}_{I,k}, \boldsymbol{v}_{I,k}, \dot{v}_{I,k})$

`// The equation can be found in Eq.3.3.`

$\boldsymbol{\omega}_B^{OI} \leftarrow A_{BO}\boldsymbol{\omega}_O^{OI}$

$\boldsymbol{\omega}_B^{BI} \leftarrow \boldsymbol{\omega}_B^{BO} + \boldsymbol{\omega}_B^{OI}$

$\boldsymbol{\omega}_{c,k} \leftarrow \boldsymbol{\omega}_B^{BI}$

`// The desired angular velocity is obtained.`

Algorithm 6 MODE 4 (MEKF + Sliding Mode Controller)

Inputs : σ_{mag}: standard deviation of the magnetometer measurement noise

σ_{sun}: standard deviation of the sun sensor measurement noise

$\boldsymbol{m}_{B,k}$: magnetometer measurement at time k

$\boldsymbol{m}_{B,k+1}$: magnetometer measurement at time k+1

$\boldsymbol{s}_{B,k}$: sun sensor measurement at time k

$\boldsymbol{m}_{R,k}$: magnetic field at ECI at time k

$\boldsymbol{s}_{R,k}$: sun direction at ECI at time k

$\boldsymbol{J}$: Moment of inertia

dt: time incrementation

$which\_filter$: 1 = MEKF with gyroscope measurements or 2 = MEKF with mathematical model. The BN can provide these inputs.

$which\_mea$: which quaternion measurement is being used, 1 = QUEST or 2 = Star tracker. The BN can provide these inputs.

$\hat{\boldsymbol{x}}_{k|k-1}$: Global states, differs according the $which\_filter$

$\Sigma_{k|k-1}$: Covariance matrix, differs according the $which\_filter$

$\tilde{\boldsymbol{\omega}}_k$: Gyroscope measurement at time k, which is used if the $which\_filter = 1$

$\tilde{\boldsymbol{q}}_{str,k}$: Star tracker measurement at time k, which is used if the $which\_mea = 2$

$\boldsymbol{h}_k$: Reaction wheel angular momentum at time k

$\dot{\boldsymbol{h}}_k$: Change at reaction wheel angular momentum at time k

$\boldsymbol{q}_{c,k}$: Commanded (desired) quaternions at time k

$\boldsymbol{\omega}_{c,k}$: Commanded (desired) angular velocity at time k

$\dot{\boldsymbol{\omega}}_{c,k}$: Commanded (desired) angular velocity time derivative at time k

Outputs: $\hat{\boldsymbol{x}}_{k|k}$: estimated global states at time k

$\hat{\boldsymbol{x}}_{k+1|k}$: estimated global states at time k+1

$\Sigma_{k|k}$: estimated covariance at time k

$\Sigma_{k+1|k}$: estimated covariance at time k+1

$\boldsymbol{L}$: commanded moment to reaction wheel

$L \leftarrow sliding\_mode(\boldsymbol{q}_{c,k}, \hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{\omega}_{c,k}, \dot{\boldsymbol{\omega}}_{c,k}, \boldsymbol{J})$ `// To use the sliding mode controller,` $\hat{\boldsymbol{\omega}}_k$ `is required. If` $which\_filter = 2$, `then` $\hat{\boldsymbol{\omega}}k$ `is included in` $\hat{\boldsymbol{x}}_{k|k-1}$.

// However, if $which\_filter = 1$, the gyro bias $\hat{\boldsymbol{b}}_{g,k}$ is included in the global states instead. In this case, $\hat{\boldsymbol{\omega}}_k$ can be calculated by subtracting $\hat{\boldsymbol{b}}_{g,k}$ from the measured angular velocity $\tilde{\boldsymbol{\omega}}_k$.

If $which\_mea = 1$ **then**

$w \leftarrow [1/\sigma_{mag}, 1/\sigma_{sun}]$

$w \leftarrow w/norm(w)$

$b \leftarrow [\boldsymbol{m}_{B,k}/\|\boldsymbol{m}_{B,k}\| \;\; \boldsymbol{s}_{B,k}/\|\boldsymbol{s}_{B,k}\|]$

$r \leftarrow [\boldsymbol{m}_{R,k}/\|\boldsymbol{m}_{R,k}\| \;\; \boldsymbol{s}_{R,k}/\|\boldsymbol{s}_{R,k}\|]$

$\tilde{\boldsymbol{q}}_k \leftarrow Quest(w, r, b)$

else

$\tilde{\boldsymbol{q}}_k \leftarrow \boldsymbol{q}_{str,k}$

end

If $which\_filter = 1$ **then**

$[\hat{\boldsymbol{x}}_{k|k}, \hat{\boldsymbol{x}}_{k+1|k}, \Sigma_{k+1|k}] \leftarrow MEKF\_1(\hat{\boldsymbol{x}}_{k|k-1}, \Sigma_{k|k-1}, \tilde{\boldsymbol{q}}_k, \tilde{\boldsymbol{\omega}}_k, dt)$

else

$[\hat{\boldsymbol{x}}_{k|k}, \hat{\boldsymbol{x}}_{k+1|k}, \Sigma_{k+1|k}] \leftarrow MEKF\_2(\hat{\boldsymbol{x}}_{k|k-1}, \Sigma_{k|k-1}, \tilde{\boldsymbol{q}}_k, \boldsymbol{L}, \boldsymbol{h}_k, \dot{\boldsymbol{h}}_k, J, dt)$

// In this mode, where the reaction wheel is utilized as an actuator, the vectors $\boldsymbol{h}_k$ and $\dot{\boldsymbol{h}}_k$ are not equal to zero.

end

6.1.5 Mode 5

This mode is specifically designed for reaction wheel desaturation purposes, as explained in Section 4.2.3. In this mode, both the magnetorquer and reaction wheel are used as actuators. MEKF is employed for the estimation algorithm since the spacecraft's angular velocity is still low. As shown in Figure 6.1, to transition to mode 5, the previous mode must be in mode 4, where the spacecraft's angular velocity is low. Even with both the magnetorquer and reaction wheel working simultaneously in mode 5, the spacecraft's angular velocity cannot increase since their total torque values are inverse of each other, i.e., $\boldsymbol{L}_{mtq} = -\boldsymbol{L}_{rw}$. Thus, the total torque applied to the spacecraft is zero (except perturbation torques), and the spacecraft's angular ve-

locity remains almost constant. The initial value of this angular velocity is coming from mode 4, where it is quite small. Thus, it is ensured that the Kalman filter will not diverge due to high angular velocity of the spacecraft. The algorithm presented in Algorithm 7 provides detailed information about the mode 5.

Algorithm 7 MODE 5 (MEKF + Reaction Wheel Desaturation Controller)

Inputs : σ_{mag}: standard deviation of the magnetometer measurement noise

σ_{sun}: standard deviation of the sun sensor measurement noise

$\boldsymbol{m}_{B,k}$: magnetometer measurement at time k

$\boldsymbol{m}_{B,k+1}$: magnetometer measurement at time k+1

$\boldsymbol{s}_{B,k}$: sun sensor measurement at time k

$\boldsymbol{m}_{R,k}$: magnetic field at ECI at time k

$\boldsymbol{s}_{R,k}$: sun direction at ECI at time k

$\boldsymbol{J}$: Moment of inertia

dt: time incrementation

$which\_filter$: 1 = MEKF with gyroscope measurements or 2 = MEKF with mathematical model. This can be externally selected.

$which\_mea$: which quaternion measurement is being used, 1 = QUEST or 2 = Star tracker. This can be externally selected.

$\hat{\boldsymbol{x}}_{k|k-1}$: Global states, differs according the $which\_filter$

$\Sigma_{k|k-1}$: Covariance matrix, differs according the $which\_filter$

$\tilde{\boldsymbol{\omega}}_k$: Gyroscope measurement at time k, which is used if the $which\_filter = 1$

$\tilde{\boldsymbol{q}}_{str,k}$: Star tracker measurement at time k, which is used if the $which\_mea = 2$

$\boldsymbol{h}_k$: Reaction wheel angular momentum at time k

$\dot{\boldsymbol{h}}_k$: Change at reaction wheel angular momentum at time k

Outputs: $\hat{\boldsymbol{x}}_{k|k}$: estimated global states at time k

$\hat{\boldsymbol{x}}_{k+1|k}$: estimated global states at time k+1

$\Sigma_{k|k}$: estimated covariance at time k

$\Sigma_{k+1|k}$: estimated covariance at time k+1

$\boldsymbol{L}_{mtq}$: commanded moment to magnetorquer

$\boldsymbol{L}_{rw}$: commanded moment to reaction wheel

$L \leftarrow saturation\_controller(\boldsymbol{h}_k, \boldsymbol{m}_{b,k})$

$\boldsymbol{L}_{mtq} \leftarrow -\boldsymbol{L}$

$\boldsymbol{L}_{rw} \leftarrow \boldsymbol{L}$

// As previously mentioned, the control law for desaturating the reaction wheel is given by $\boldsymbol{L} = [\boldsymbol{m}\times]\boldsymbol{B}$.

// To achieve this, the negative of $\boldsymbol{L}$ must be applied from magnetorquer, while the positive one should be given as the commanded moment to the reaction wheel.

If $which\_mea = 1$ **then**

$w \leftarrow [1/\sigma_{mag}, 1/\sigma_{sun}]$

$w \leftarrow w/norm(w)$

$b \leftarrow [\boldsymbol{m}_{B,k}/\|\boldsymbol{m}_{B,k}\| \;\; \boldsymbol{s}_{B,k}/\|\boldsymbol{s}_{B,k}\|]$

$r \leftarrow [\boldsymbol{m}_{R,k}/\|\boldsymbol{m}_{R,k}\| \;\; \boldsymbol{s}_{R,k}/\|\boldsymbol{s}_{R,k}\|]$

$\tilde{\boldsymbol{q}}_k \leftarrow Quest(w, r, b)$

else

$\tilde{\boldsymbol{q}}_k \leftarrow \boldsymbol{q}_{str,k}$

end

If $which\_filter = 1$ **then**

$[\hat{\boldsymbol{x}}_{k|k}, \hat{\boldsymbol{x}}_{k+1|k}, \Sigma_{k+1|k}] \leftarrow MEKF\_1(\hat{\boldsymbol{x}}_{k|k-1}, \Sigma_{k|k-1}, \tilde{\boldsymbol{q}}_k, \tilde{\boldsymbol{\omega}}_k, dt)$

else

$[\hat{\boldsymbol{x}}_{k|k}, \hat{\boldsymbol{x}}_{k+1|k}, \Sigma_{k+1|k}] \leftarrow MEKF\_2(\hat{\boldsymbol{x}}_{k|k-1}, \Sigma_{k|k-1}, \tilde{\boldsymbol{q}}_k, \boldsymbol{L}, \boldsymbol{h}_k, \dot{\boldsymbol{h}}_k, \boldsymbol{J}, dt)$

end

6.1.6 Mode 6

The hybrid mode described in this section serves two main purposes: it acts as a backup mode for emergency situations and as a sun-pointing mode for energy generation. Unlike the previous modes, the estimation algorithm may not always be necessary in this mode. Instead of using the QUEST algorithm, a star tracker can be employed. Furthermore, both the reaction wheel and magnetorquer can be utilized as actuators in this mode. To regulate the system, a sliding-mode controller is employed. To use the sliding-mode controller, the desired quaternion and angular velocities need to be determined. Algorithm 8 outlines the steps to obtain the desired quaternion and angular velocities for the sun-pointing case. The key point in this algorithm is aligning the satellite body frame component $\boldsymbol{e}_3$ with the known sun vector in the ECI frame. By using a 313 rotation sequence between the LVLH and body frames, the desired Euler angles are computed. These steps are provided as comments in Algorithm 8. Since this is a regulation problem, the desired angular velocities are set to zero. For

a comprehensive understanding of this mode, refer to Algorithm 9, which provides a detailed description of the entire process.

Algorithm 8 Desired Angular Velocity for the Sun-Pointing Case

Inputs : $\boldsymbol{r}_{I,k}$: position of the spacecraft resolved at the inertial frame at time k

$\boldsymbol{v}_{I,k}$: velocity of the spacecraft resolved at the inertial frame at time k

$\boldsymbol{s}_I$: Sun position vector resolved at the ECI frame

Outputs: $\boldsymbol{q}_{c,k}$: desired quaternion at time k

$\boldsymbol{\omega}_{c,k}$: desired angular velocity of the spacecraft with respect to inertial frame, resolved at the body frame at time k

$A_{IO} \leftarrow \begin{bmatrix} \boldsymbol{o}_{1I} & \boldsymbol{o}_{2I} & \boldsymbol{o}_{3I} \end{bmatrix}$

$A_{OI} \leftarrow A_{IO}^T$

// Transformation matrix from LVLH to inertial frame can be obtained by using the Eq.3.1.

$\boldsymbol{s}_O \leftarrow A_{OI} s_I$

$\theta = acos(\boldsymbol{s}_O(3))$

$\phi = atan2(\boldsymbol{s}_O(1), -\boldsymbol{s}_O(2))$

$\psi =$ free parameter

// $\boldsymbol{s}_I = A_{IO} A_{OB} \boldsymbol{e}_3 \Rightarrow A_{OI} \boldsymbol{s}_I = A_{OB} \boldsymbol{e}_3 \Rightarrow s_O = \begin{bmatrix} s\phi s\theta \\ -c\phi s\theta \\ c\theta \end{bmatrix}$

$A_{BO} \leftarrow A(\boldsymbol{e}_3, \psi) A(\boldsymbol{e}_1, \theta) A(\boldsymbol{e}_3, \phi)$

// Transformation from the LVLH to body frame obtained with 313 rotation. The equation can be found in Equation 3.5.

$A_{BI} \leftarrow A_{BO} A_{OI}$

$\boldsymbol{q}_{c,k} \leftarrow quat2dcm(A_{BI})$

// The desired quaternion is obtained.

$\boldsymbol{\omega}_B^{BI} \leftarrow \mathbf{0}_{3\times3}$

$\boldsymbol{\omega}_{c,k} \leftarrow \boldsymbol{\omega}_B^{BI}$

// The desired angular velocity is obtained. Since this is regulation problem angular velocity is zero vector.

Algorithm 9 MODE 6 (Sliding-Mode Controller)

Inputs : σ_{mag}: standard deviation of the magnetometer measurement noise

σ_{sun}: standard deviation of the sun sensor measurement noise

$\boldsymbol{m}_{B,k}$: magnetometer measurement at time k

$\boldsymbol{m}_{B,k+1}$: magnetometer measurement at time k+1

$\boldsymbol{s}_{B,k}$: sun sensor measurement at time k

$\boldsymbol{m}_{R,k}$: magnetic field at ECI at time k

$\boldsymbol{s}_{R,k}$: sun direction at ECI at time k

$\boldsymbol{J}$: Moment of inertia

$which\_mea$: which quaternion measurement is being used, 1 = QUEST or 2 = Star tracker. This can be externally selected.

$\tilde{\boldsymbol{\omega}}_k$: Gyroscope measurement at time k, which is used if the selected filter is equal to 1

$\tilde{\boldsymbol{q}}_{str,k}$: Star tracker measurement at time k, which is used if the selected measurement is equal to 2

$\boldsymbol{q}_{c,k}$: Commanded (desired) quaternions at time k

$\boldsymbol{\omega}_{c,k}$: Commanded (desired) angular velocity at time k

$\dot{\boldsymbol{\omega}}_{c,k}$: Commanded (desired) angular velocity time derivative at time k

Outputs: $\boldsymbol{L}$: commanded moment to magnetorquer or reaction wheel

If $which\_mea = 1$ **then**

$w \leftarrow [1/\sigma_{mag}, 1/\sigma_{sun}]$

$w \leftarrow w/norm(w)$

$b \leftarrow [\boldsymbol{m}_{B,k}/\|\boldsymbol{m}_{B,k}\| \;\; \boldsymbol{s}_{B,k}/\|\boldsymbol{s}_{B,k}\|]$

$r \leftarrow [\boldsymbol{m}_{R,k}/\|\boldsymbol{m}_{R,k}\| \;\; \boldsymbol{s}_{R,k}/\|\boldsymbol{s}_{R,k}\|]$

$\tilde{\boldsymbol{q}}_k \leftarrow Quest(w, r, b)$

else

$\tilde{\boldsymbol{q}}_k \leftarrow \boldsymbol{q}_{str,k}$

end

$L \leftarrow sliding\_mode(\boldsymbol{q}_{c,k}, \tilde{\boldsymbol{q}}_k, \boldsymbol{\omega}_{c,k}, \dot{\boldsymbol{\omega}}_{c,k}, \tilde{\boldsymbol{\omega}}_k, \boldsymbol{J})$

6.2 Mode Transition with Thresholds

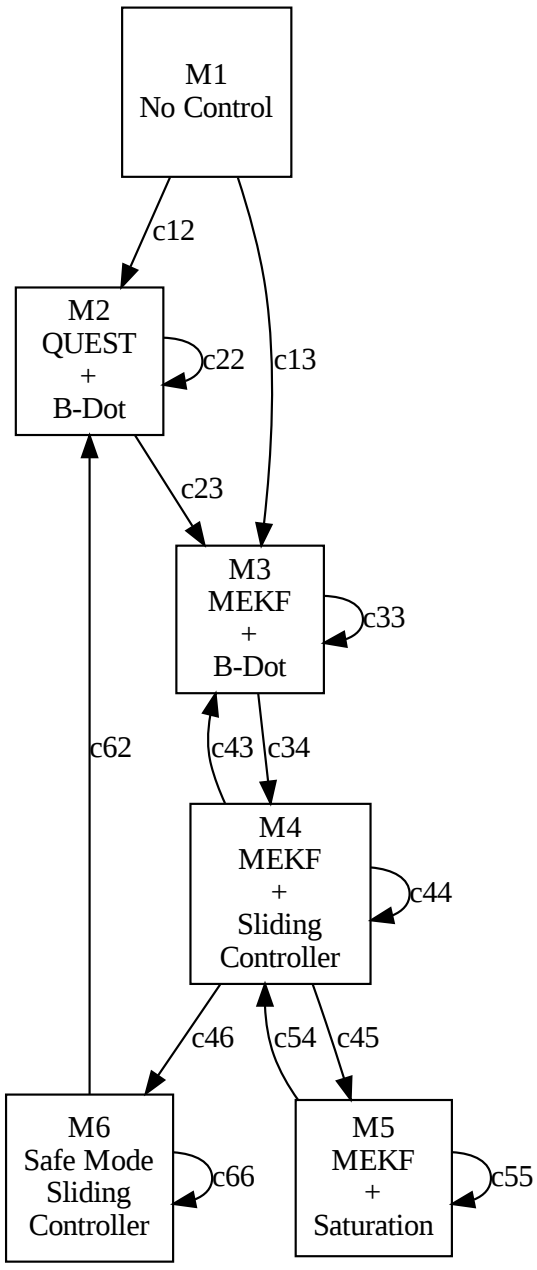


Figure 6.1: Mode transition thresholds

In this section, the transitions between different modes of the hybrid algorithms are explained using threshold values. This subsection is an important part of the thesis as it describes the operation of hybrid algorithms. Figure 6.1 illustrates the transitions between different modes with arrows, which indicate the direction of the transition. The labels on the arrows are subscripts of the condition variable c, where the first element represents the incoming node, and the second element represents the outgoing node. The explanation of why mode transitions are depicted in this manner is presented first. Subsequently, all the conditions for the transitions are explained one-by-one to provide a complete understanding of the mode transitions in the hybrid algorithms.

As previously discussed, the first mode of the algorithm serves as the starting mode. Then, based on certain conditions, the spacecraft transitions to either mode 2 or mode 3. Both mode 2 and mode 3 share a common algorithm, which is the use of the B-dot controller to reduce the spacecraft's angular velocity. This process is referred to as detumbling.

Once the spacecraft's angular velocity has decreased, a sliding-mode controller and MEKF are utilized. This mode is the primary mission mode and is connected to both mode 5 and mode 6. Mode 5 is characterized by the utilization of the reaction wheel desaturation controller, while mode 6 serves as a safe mode.

Next, each of the conditions that initiate the mode transitions will be discussed in a sequential manner.

- $c_{12} = \|\tilde{\boldsymbol{\omega}}_k\| \geq \omega_{thr}$
  $c_{13} = \|\tilde{\boldsymbol{\omega}}_k\| < \omega_{thr}$
  where
  $\omega_{thr} = 1.5$ [deg/s]
- $c_{22} = \|\tilde{\boldsymbol{\omega}}_k\| \geq \omega_{thr}$
  $c_{23} = \|\tilde{\boldsymbol{\omega}}_k\| < \omega_{thr}$
  where
  $\omega_{thr} = 1.5$ [deg/s]

The conditions for mode transitions, namely c_{12}, c_{13}, c_{22}, and c_{23}, have been given above. These conditions are dependent on the angular velocity of the spacecraft, and the threshold value ω_{thr} can be selected based on the specific spacecraft requirements. Additionally, it is important to note that in the sequence of 20 samples, the conditions must be met for the 10 samples. This is crucial as abrupt measurements can sometimes occur at the sensor, and meeting the conditions for the first 10 samples helps prevent this issue. If the conditions for mode transitions are not met within 20 samples, then the indexing returns to 1 and the sequence of conditions is restarted. This ensures that the conditions are continuously monitored and transitions occur only when the necessary conditions are met.

Before presenting the conditions c_{33} and c_{34}, it is necessary to provide some preliminary information.

1. **Confidence Interval**
   Suppose we have a set of random variables $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$, each taken from a distribution characterized by an unknown parameter $\boldsymbol{\theta}$. Rather than directly

estimating the value of $\boldsymbol{\theta}$, we may instead estimate an interval for it. This is because we cannot know how close our estimate $\hat{\boldsymbol{\theta}}$ is to the true value of $\boldsymbol{\theta}$ without additional information [1]. Estimating a lower bound $\hat{\boldsymbol{\theta}}_l$ and an upper bound $\hat{\boldsymbol{\theta}}_u$ for the parameter $\boldsymbol{\theta}$ is therefore more informative. The confidence level is a critical parameter when estimating a interval. In Equation 6.1, the symbol $1 - \alpha$ represents the confidence level. An example can illustrate the relationship between confidence level and probability. Suppose the confidence level is set to 95%, or equivalently, $\alpha = 0.05$. Then, the probability that the true value of the parameter falls within the calculated interval is 0.95, which corresponds to the right-hand side of Eq. 6.1.

$$P(\hat{\boldsymbol{\theta}}_l \leq \boldsymbol{\theta} \leq \hat{\boldsymbol{\theta}}_u) \geq 1 - \alpha \tag{6.1}$$

2. **Chi-Squared Distributions**

   Let $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$ be independent random variables with a standard normal distribution, denoted as $\boldsymbol{X} \sim \mathcal{N}(0, 1)$. In this case, a new random variable can be defined as shown in Eq. 6.2.

   $$\boldsymbol{Y} = \boldsymbol{X}_1 + \boldsymbol{X}_2 + \cdots + \boldsymbol{X}_n \tag{6.2}$$

   Since the addition of independent Gaussian variables results in a Gaussian distribution, the sum $\boldsymbol{Y}$ of the random variables $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$ also has a Gaussian distribution. Another random variable $\boldsymbol{Z}$ can be defined as the sum of the squared of the standard normal random variables, which follows a chi-squared distribution with n degrees of freedom denoted as $\boldsymbol{Z} \sim \tilde{\chi}^2(n)$. This is represented in Eq. 6.3.

   $$\boldsymbol{Z} = \boldsymbol{X}_1^2 + \boldsymbol{X}_2^2 + \cdots + \boldsymbol{X}_n^2 \tag{6.3}$$

   This distribution is important due to its close relationship with Gaussian distributions.

3. **Innovation in Kalman Filter**

   Innovation refers to the new information obtained from a measurement that is different from the estimated value of $E[\boldsymbol{y}_k \mid \boldsymbol{y}_{0:k-1}]$, as illustrated in Eq. 6.4. In the Kalman Filter, the innovation $\boldsymbol{\nu}_k$ is multiplied by the Kalman gain to obtain the correction term for the state estimate.

   $$\boldsymbol{\nu}_k = \tilde{\boldsymbol{y}}_k - E[\boldsymbol{y}_k \mid \boldsymbol{y}_{0:k-1}] \tag{6.4}$$

Jointly Gaussian random variables retain their Gaussian property even when transformed by affine functions. Therefore, the innovation sequence, which is derived from the difference between the observed measurements and their predicted values in the Kalman Filter, can also be assumed to be Gaussian.

The zero-mean property of the innovation sequence $\boldsymbol{\nu}_k$ for $k = 0, 1, \ldots$ can be shown by using the iterated expectations.

$$\begin{aligned} \bar{\boldsymbol{\nu}}_k &= E[\tilde{\boldsymbol{y}}_k - E[\boldsymbol{y}_k \mid \boldsymbol{y}_{0:k-1}]] \\ &= \bar{\boldsymbol{y}}_k - E[E[\boldsymbol{y}_k \mid \boldsymbol{y}_{0:k-1}]] \\ &= \bar{\boldsymbol{y}}_k - \bar{\boldsymbol{y}}_k \\ &= 0 \end{aligned} \tag{6.5}$$

Another property of the innovation sequence is that it is an independent sequence. This means that the innovations at different time steps are uncorrelated. Since independence implies uncorrelatedness, Eq. 6.6 can be written as:

$$\begin{aligned} E[\boldsymbol{\nu}_i^T \boldsymbol{\nu}_j] &= E[\boldsymbol{\nu}_i^T] E[\boldsymbol{\nu}_j] \;\; \forall i \neq j \\ &= 0 \cdot 0 \\ &= 0 \end{aligned} \tag{6.6}$$

As previously stated, the innovation sequence $\boldsymbol{\nu}_k$ for $k = 0, 1, \ldots$ is independent and has zero mean. This allows for the definition of a new variable which follows a Chi-Squared distribution. Specifically, a new variable can be defined as the normalized innovation squared, as shown in Equation 6.7.

$$\epsilon_{\nu,k} = \boldsymbol{\nu}_k^T S(k)^{-1} \boldsymbol{\nu}_k \tag{6.7}$$

where

$S_k =$ Innovation covariance of the Kalman filter, $H \Sigma_{k|k-1} H^T + R$

The motivation behind normalizing the innovation squared with the innovation covariance S_k is to obtain a random variable in the same form as $\boldsymbol{Y}$, as shown in Equation 6.2. Therefore, a new random variable $\boldsymbol{K}$ can be defined as shown in Equation 6.8. Then, by squaring this random variable and taking the summation, as depicted in Equation 6.3, the resulting variable can be normalized by dividing

it by the innovation covariance S_k, yielding the normalized innovation squared given in Equation 6.7.

$$\boldsymbol{K} = \sum_{i=0}^{k} S_i^{-1/2} \boldsymbol{v}_i \tag{6.8}$$

The innovation of the measurement can be used as a means of analyzing the consistency of the filter, which is indicative of whether the filter is functioning properly or not. This is because a finite number of measurements should be consistent with their theoretical properties. It is recommended a more comprehensive understanding of the topics discussed should refer to the book cited as [10].

- $c_{33} = \epsilon_{v,k} \geq \epsilon_{v,thr}$
  $c_{34} = \epsilon_{v,k} < \epsilon_{v,thr}$

where

$$\epsilon_{v,thr} = 7.81$$

Given that the normalized innovation squared has three degrees of freedom and assuming a $\alpha = 0.05$, the upper bound of the confidence interval is set to $\epsilon_{v,thr} = 7.81$. Note that the lower bound check is not performed in this case, but if it were to be performed, the value would be equal to 0.35. To prevent impulsive changes in the sequence of 20 samples, it is required that the conditions are satisfied for the first 10 samples. If the conditions are not met within the first 20 samples, the sample counter will be reset to 1.

•

Table 6.1: Transition condition from mode 4

$\|\boldsymbol{h}_k\| \geq \boldsymbol{h}_{thr}$	Index=Index$_{thr}$	$\epsilon_{v,k} \geq \epsilon_{v,thr}$	Condition
0	0	0	$c_{44} = 1$
0	0	1	
0	1	0	
0	1	1	$c_{46} = 1$
1	0	0	$c_{45} = 1$
1	0	1	
1	1	0	
1	1	1	

As depicted in Table 6.1, the process of transitioning from mode 4 is a complicated one that encompasses several conditions. The first condition depends on the angular momentum of the reaction wheel and whether it exceeds a certain threshold. This is a critical parameter for transitioning to mode 5.

where

$h_{thr} = 0.88$ [Nms]

The second condition Index represents the cycle number for the BN and ranges from 0 to a maximum of 2. For each cycle, the Index value is incremented by 1. For instance, if the condition $\Big(\big(\|\boldsymbol{h}_k\| < \boldsymbol{h}_{thr}\big) \wedge \big(\epsilon_{v,k} \geq \epsilon_{v,thr}\big) \wedge \big(Index = 0\big)\Big) = 1$ is met, and graph or filter#1 is used, the BN conducts probabilistic calculations and selects graph or filter#2. If this approach is unsuccessful, the BN is used again, and the result indicates that graph or filter#1 should be used again. This means that one cycle has been completed, and the Index value is increased by 1. The new condition becomes $\Big(\big(\|\boldsymbol{h}_k\| < \boldsymbol{h}_{thr}\big) \wedge \big(\epsilon_{v,k} \geq \epsilon_{v,thr}\big) \wedge \big(Index = 1\big)\Big) = 1$. Instead of transitioning

directly to mode 6, the BN addresses the problem in mode 4 again. If this approach fails, another cycle is completed, and the Index value is increased to 2. The transition from mode 4 to mode 6 occurs only after, $\Big(\big(\|\boldsymbol{h}_k\| < \boldsymbol{h}_{thr} \big) \wedge \big(\epsilon_{v,k} \geq \epsilon_{v,thr} \big) \wedge \big(Index = 1 \big) \Big) = 2$.

where

$$Index_{thr} = 2$$

The third parameter, the normalized innovation squared parameter, works on the same principle as in mode 3. Table 6.1 provides a summary of the transition conditions from mode 4 to mode 4, 5 and 5.

where

$$\epsilon_{v,thr} = 7.81$$

- $c_{54} = \|\boldsymbol{h}_k\| < h_{min}$
- $c_{55} = \|\boldsymbol{h}_k\| \geq h_{min}$

where

$$h_{min} = 0.088 \text{ [Nms]}$$

When switching from mode 5 to another mode, the condition for the transition is to achieve the minimum angular momentum for the reaction wheel. This minimum value is the remaining angular momentum in the reaction wheel, which may differ depending on the specific characteristics of the reaction wheel being utilized.

- $c_{62} = LeaveSafeMode$
- $c_{66} = LeaveS\overline{afe}Mode$

The satellite cannot transition from safe mode to other operational modes unless an external command is received from the ground-station. Specifically, transitioning from mode 6 to mode 2 can only occur when the $LeaveSafeMode$ command is issued from the ground-station. This command triggers a direct transition to mode 2. It's important to note that the logical not of the $LeaveSafeMode$ command, which is represented by c_{66}, also affects the transition process.

6.3 Mode Transition with Bellman Optimization

In this context, an alternative approach is employed for mode transitions, deviating from direct usage of threshold values. Since the spacecraft must operate in one of the six modes described in Section 6.2, these modes can be seen as the state-space representation of mode transition. Furthermore, since the modes can be discretely defined, it implies that there is a finite number of modes and their dynamics, i.e., changes over time, can be modeled using a Markov chain. Moreover, to exert control over the modes, i.e., to achieve the desired mode transitions, controlled Markov chains are utilized. Optimal control of the Markov chain is achieved through Bellman optimization.

In this part, cost functions are created based on the threshold values used in Section 6.2. This approach serves as a complementary extension to Section 6.2, where Markov chains were used in a deterministic manner, in contrast to their inherent stochastic nature. As a whole, this part can be seen as another representation of mode transitions with thresholds, where Markov chains are used deterministically and cost functions are created based on the threshold values. The difference lies in the methodology used to achieve the mode transitions.

To begin with, a brief discussion is provided on the concepts of a finite state Markov chain, transition probability matrix, controlled Markov chain, cost function, and Bellman optimization. These concepts lay the foundation for understanding the subsequent explanations. Finally, the application of these concepts in the context of this thesis is presented. The specific ways in which the concepts are employed and their implications within the thesis are described. Additionally, it is worth noting that in the future work section, a more detailed discussion is provided, highlighting potential

areas for further exploration and improvement related to the concepts discussed in this section.

- **Finite State Markov Chains** A finite state Markov chain consists of a finite number of states, represented as $x_k \in 1, 2, 3, \ldots, I$, where the subscript k denotes the time step k. The transition between states can be expressed as the state at time $k+1$ being a function of the state at time k, as well as the associated noise. This mathematical representation is given in Eq. 6.9 using the 1-step transition Markov property. The randomness or stochasticity in the Markov chain arises from this associated noise, and the uncertainty in knowing the exact initial condition is another source of randomness. For readers seeking introductory knowledge on the subject, valuable sources include [24, 52]. However, for those who require more detailed information, [23, 55] would be more suitable.

$$P(\boldsymbol{x}_{k+1} \mid \boldsymbol{x}_{0:k}) = P(\boldsymbol{x}_{k+1} \mid \boldsymbol{x}_k) \tag{6.9}$$

- **Transition Probability Matrix (TPM)** A Markov chain's transition probability matrix (TPM) can be defined as a matrix, as the name suggests, as shown in Eq. 6.10. Again, the subscript k denotes the time step. This means that different TPM can be available at different time.

$$P_k = \begin{bmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1I} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2I} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ p_{I1} & p_{I2} & p_{I3} & \cdots & p_{II} \end{bmatrix} \tag{6.10}$$

where

$$p_{ij}(k) = P(x_{k+1} = j \mid x_k = i)$$

The matrix P_k referred to in this context is known as a stochastic matrix. A stochastic matrix is defined as a square matrix in which each row represents a probability vector. Specifically, in Equation 6.11, it can be observed that first row of the matrix P_k corresponds to the probability vector for state 1 at time k.

$$p_k^1 = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1I} \end{bmatrix} \tag{6.11}$$

According to the definition of probability, the elements of a probability vector must be greater than or equal to zero, i.e., $p_{ij} \geq 0$ for $1 \leq i, j \leq I$. Furthermore, the sum of the elements in a probability vector must be equal to one. This implies that the sum of the rows of the TPM must be equal to one, i.e., $\sum_{j=1}^{I} p_{ij} = 1$ for $i = 1, \ldots, I$.

- **Controlled Markov Chain** In the context of controlled Markov chains, the transition between states at time $k+1$ is not only influenced by the state at time k and associated noise, but also by the input at time k, denoted as u_k. This relationship can be expressed mathematically as shown in Eq. 6.12. By appropriately selecting and applying the input, it becomes possible to control the transitions within the Markov chain. This implies that different TPMs are defined for different inputs. The work conducted by Kumar et al. [35] provides a framework for analyzing controlled Markov chains, offering insights into the dynamics and behavior of such systems.

$$P(\boldsymbol{x}_{k+1} \mid \boldsymbol{u}_{0:k}, \boldsymbol{x}_{0:k}) = P(\boldsymbol{x}_{k+1} \mid \boldsymbol{u}_k, \boldsymbol{x}_k) \tag{6.12}$$

- **Cost Function** The cost function plays a crucial role in controlling the transitions between states within the Markov chain. Its purpose is to determine the optimal input that leads to the desired transitions. In general, the cost function is a function of the state and input at time k, denoted as x_k and u_k respectively. Mathematically, the cost function can be represented as $c_k(x_k, u_k)$. The specific form of the cost function depends on the problem at hand and the desired objectives of the control system. In addition, two important concepts related to cost are the total cost and incremental cost. The incremental cost, denoted as $c_k(x_k, u_k)$, represents the cost at a specific time step k, taking into account the state x_k and the input u_k at that time. On the other hand, the total cost is the cumulative sum of incremental costs over a specified time horizon, from $k = 0$ to $k = N$. It is given by $\sum_{k=0}^{N-1} c_k(x_k, u_k) + c_N(x_k)$, where $c_N(x_k)$ represents the cost at time N that is solely dependent on the state x_N and not influenced by the input u_N. This distinction arises because the input at time N can impact the subsequent states beyond time N, while the problem horizon is limited to N and does not extend further. However, due to the stochastic nature of Markov chains, the expression $\sum_{k=0}^{N-1} c_k(x_k, u_k) + c_N(x_k)$ may not provide a

comprehensive understanding of the cost. To obtain a more meaningful cost measure, we can define the cost function as the expectation of the cumulative cost over the time horizon. Therefore, a more applicable cost definition is given by $J = E\left[\sum_{k=0}^{N-1} c_k(x_k, u_k) + c_N(x_k)\right]$, where $E[\cdot]$ represents the expectation operator.

- **Bellman Optimization** Bellman Optimization is a recursive approach used to determine the optimal input for a controlled Markov chain. The optimization process is conducted in a backward manner due to the fact that the input at time step k, denoted as u_k, influences the system dynamics and transitions beyond time k, specifically at time steps $k+1$, $k+2$, and so on. A fundamental concept in understanding Bellman optimization is the cost-to-go function. Mathematically, the cost-to-go is defined as shown in Eq. 6.13. It is important to note that in previous sections, the incremental cost was defined as $c_k(x_k, u_k)$, whereas in Eq. 6.13, it is defined as $c_k(x_k, g_k(x_k))$. This adjustment reflects the closed-loop control nature, where the input is a function of the state x_k. Returning to the definition of the cost-to-go, it can be viewed as the summation of the incremental costs from time step k to N, given the known state at time step k, denoted as x_k.

$$V_k(i) = E\left[\sum_{l=k}^{N} c_l(x_l, g_l(x_l))\Big| x_k = i\right] \tag{6.13}$$

In Eqs. 6.14 and 6.15, a summary of the cost-to-go recursion is presented. The specific details of the recursion process are not discussed here, but for a more comprehensive understanding, reference [35] can be consulted. It provides in-depth information and explanations regarding the cost-to-go recursion and its implementation.

$$V_N(x_N) = c_N(x_N) \tag{6.14}$$

$$V_k(x_k) = c_k(x_k, g_k(x_k)) + P_k V_{k+1}(x_{k+1}) \tag{6.15}$$

For optimal control in finite state Markov chains, Eqs. 6.14 and 6.15 are almost identical. The key difference lies in the definition of $V_k^\star(x_k)$, which represents the optimal cost-to-go. Eq. 6.16 provides the definition for $V_k^\star(x_k)$. To achieve optimal control, Eqs. 6.17 and 6.18 are used. For a detailed derivation of these formulas and a comprehensive understanding of the intermediate steps, refer-

ence [35] again can be consulted.

$$V_k^\star(x_k) = \min_{u_{k:N}} V_k(x_k) \tag{6.16}$$

$$V_N^\star = V_N = c_N(x_N) \tag{6.17}$$

$$V_k^\star(x_k) = \min_{u_k} \left(c_k(x_k, g_k(x_k)) + P_k(u_k) V_{k+1}^\star(x_{k+1}) \right) \tag{6.18}$$

6.3.1 Application

In this application, the state-space of the spacecraft is defined using three parameters: the norm of the angular momentum ($\|\boldsymbol{h}\|$), the normalized innovation squared of the Kalman Filter (ϵ_ν), and the norm of the angular velocity ($\|\boldsymbol{\omega}\|$). To handle these continuous variables, discretization is performed by defining high and low values for each parameter, such as $\|\boldsymbol{h}\|_{high}$ and $\|\boldsymbol{h}\|_{low}$. Figure 6.2 illustrates the discrete state Markov chain at time k and $k+1$. These states are constructed based on the previous mode transitions, which is threshold values. For instance, when examining mode 2, the application of QUEST and B-dot algorithms defines this mode. However, the question arises: How can we define mode 2 using parameters such as $\|\boldsymbol{h}\|$, ϵ_ν, and $\|\boldsymbol{\omega}\|$? This aspect of mode transition is tailored to this specific consideration, representing modes with discretized parameters. It's important to note that this discretization is only applied to mode 2, mode 3, mode 4, and mode 5. The reason behind this choice is that mode 1 is only used for initializing the algorithm, and mode 6's transition relies on ground station decision. In summary, in Figure 6.2, state 1 corresponds to mode 2, state 2 corresponds to mode 3, state 3 corresponds to mode 4, and state 4 corresponds to mode 5. In fact, in state 3, there is no information about the angular velocity, which means that it can be either high or low. Similarly, in state 4, both the normalized innovation squared and angular velocity can take on both high and low. Furthermore, it's important to consider that the selection of algorithms has a direct impact on the discretization of these states. For instance, the B-dot algorithm can influence whether the angular velocity is high ($\|\boldsymbol{\omega}\|_{\text{high}}$) or low ($\|\boldsymbol{\omega}\|_{\text{low}}$). These algorithms, which are estimators and controllers, are taken as input parameters in this process.

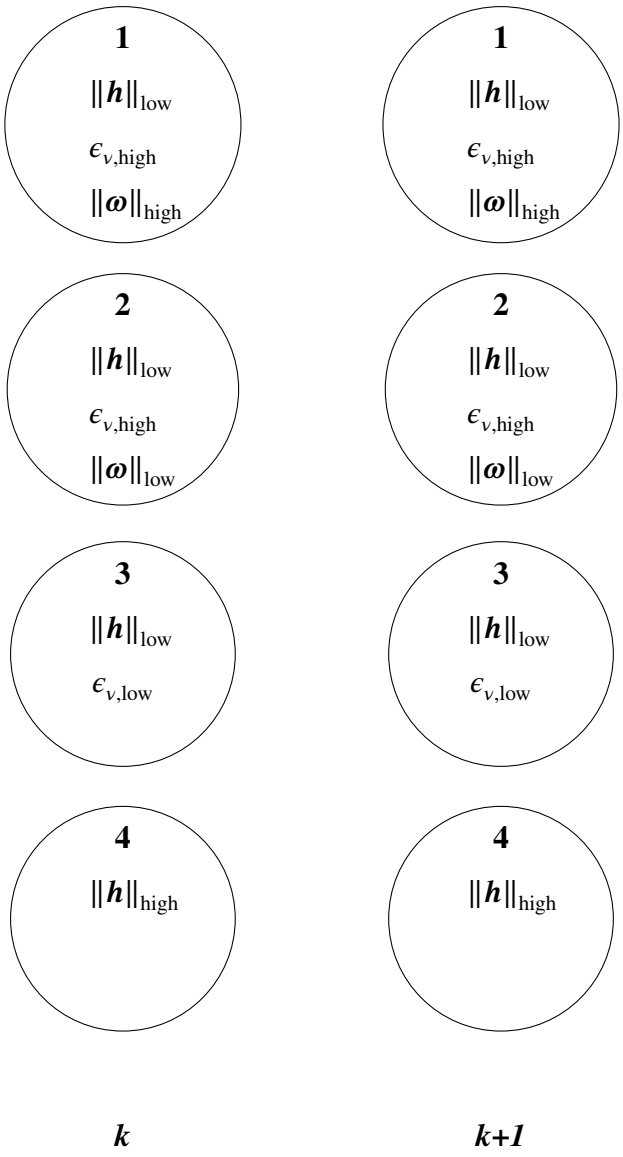


Figure 6.2: Discrete state Markov chain represents the spacecraft mathematical model

There are four different inputs in the system, which can be interpreted as estimators and controllers as shown in Figure 6.1. Each mode has unique properties associated with it. The inputs can be listed as follows:

$u_k \in \{QUEST\ and\ B-dot, MEKF\ and\ B-dot, MEKF\ and\ Sliding-mode, MEKF\ and\ Reaction\ Wheel\ Desaturation\}$

- $u_k = QUEST\ and\ B-dot$: Input applied at high angular velocities ($\|\boldsymbol{\omega}\|_{\text{high}}$)
- $u_k = MEKF\ and\ B-dot$: Input applied at low angular velocities ($\|\boldsymbol{\omega}\|_{\text{low}}$)

- $u_k = MEKF\ and\ Sliding-mode$: Input applied at low normalized innovation squared ($\epsilon_{\nu,\text{low}}$) and low angular momentum ($\|\boldsymbol{h}\|_{\text{low}}$)
- $u_k = MEKF\ and\ Reaction\ Wheel\ Desaturation$: Input applied at high angular momentum ($\|\boldsymbol{h}\|_{\text{high}}$)

TPMs are constructed deterministically using the experience gained from transitions documented in the literature [36, 45], as well as the experience obtained from performing transitions using threshold values, as described in section 6.2.

$$P(u_k = \text{QUEST and B-dot}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.19}$$

$$P(u_k = \text{MEKF and B-dot}) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.20}$$

$$P(u_k = \text{MEKF and Sliding-mode}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{6.21}$$

$$P(u_k = \text{MEKF and Reaction Wheel Desaturation}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.22}$$

The cost is calculated based on the values of the angular velocity, angular momentum, and normalized innovation squares at time k. The interpretation of what is considered a "high" state determines how the cost is computed. This means that the specific thresholds for defining a "high" state influence the cost calculation. These values,

which can be obtained from sensors or calculated, are used to determine the cost according to the given criteria for a "high" state.

$$c_k(x_k) = \begin{cases} c_k(1), & x_k = 1 \\ c_k(2), & x_k = 2 \\ c_k(3), & x_k = 3 \\ c_k(4), & x_k = 4 \end{cases} \tag{6.23}$$

where

$$c_k(1) = \left(1 - I(\|\boldsymbol{h}_k\| < h_{max,thr})\right) + \left(1 - I(\epsilon_{v,k} \geq \epsilon_{v,thr})\right) + \left(1 - I(\|\boldsymbol{\omega}_k\| \geq \omega_{thr})\right)$$

$$c_k(2) = \left(1 - I(\|\boldsymbol{h}_k\| < h_{max,thr})\right) + \left(1 - I(\epsilon_{v,k} \geq \epsilon_{v,thr})\right) + \left(1 - I(\|\boldsymbol{\omega}_k\| < \omega_{thr})\right)$$

$$c_k(3) = \left(1 - I(\|\boldsymbol{h}_k\| < h_{max,thr})\right) + \left(1 - I(\epsilon_{v,k} < \epsilon_{v,thr})\right)$$

$$c_k(4) = \begin{cases} \left(1 - I(\|\boldsymbol{h}_k\| \geq h_{max,thr}\right), & x_k \neq 4 \\ \left(1 - I(\|\boldsymbol{h}_k\| \geq h_{min,thr}\right), & x_k = 4 \end{cases}$$

The function $I(\cdot)$ is an indicator function that outputs a value of 1 if the logical variable inside the function is true, and 0 otherwise. In the context of the cost function, since all modes are observable, the cost $c_k(4)$ changes depending on whether $x_k \neq 4$ or $x_k = 4$. Also, all of threshold values are taken according to section 6.2.

CHAPTER 7

RESULTS

This chapter presents the results. Table 7.1 provides a summary of all the parameters used in the simulations.

Section 7.1 provides an overview of the performance of various algorithms described in the thesis. These algorithms include QUEST, B-dot controller, MEKF with gyroscope measurements, MEKF with mathematical model, sliding-mode controller, and reaction wheel desaturation controller.

Section 7.2 presents the performance of the ADCS. This section provides a demonstration of how the estimators, controllers, and modes operate together in a coordinated manner. Additionally, the mode transition process is explored using two different methods: threshold values and Bellman optimization. The hybrid mode performance is analyzed in two parts based on the method employed during the transition.

Lastly, in Section 7.3, the results of the FDI algorithms, specifically the BN results, are presented.

Table 7.1: Satellite parameters, initial conditions, and MEKF parameters

	Parameter	**Value**
Satellite	Mass	95 [kg]
	Inertia Matrix	$J = \begin{bmatrix} 7.84 & 0.12 & 0 \\ 0.12 & 6.90 & 0.11 \\ 0 & 0.11 & 7.58 \end{bmatrix}$ [kgm^2]

Continued on next page

Table 7.1 – Continued from previous page

	Parameter	**Value**
reaction wheel	Max. Angular Momentum	0.88 [Nms] ($I = 1.4 \times 10^{-3}$ [kgm^2], $\omega = 6000$ [rpm])
	Min. Angular Momentum	0.088 [Nms]
Magnetorquer	Max. Dipole Moment	3 [Am^2]
Sun Sensor	Standard Deviation (Normalized)	0.002
Magnetometer	Standard Deviation (Normalized)	0.02
Gyroscope	Standard Deviation	0.002 [deg/s]
	Bias	5×10^{-3} [deg/s]
Star Tracker	Standard Deviation	40 [arcsec]
Sliding-mode Controller	λ	0.015
	G	$0.15 I_{3\times3}$
	ϕ	0.01
Orbit Parameters	Radius	705 [km]
	Eccentricity	0
	Inclination	98 [deg]
Initial Condition	$\boldsymbol{q}$	$\begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$
	$\boldsymbol{\omega}$	$\begin{bmatrix} 0.05 & 0.03 & 0.04 \end{bmatrix}^T [rad/s]$
MEKF with Gyroscope Measurements	Q	$\begin{bmatrix} 10^{-7} I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 10^{-11} I_{3\times3} \end{bmatrix}$
	R	$0.04 I_{3\times3}$
	Σ_0	$\begin{bmatrix} 10^{-3} I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 10^{-5} I_{3\times3} \end{bmatrix}$
MEKF with Mathematical Model	Q	$\begin{bmatrix} 10^{-5} I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0.15 \times 10^{-6} I_{3\times3} \end{bmatrix}$
	R	$0.04 I_{3\times3}$
	Σ_0	$\begin{bmatrix} 10^{-2} I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 10^{-4} I_{3\times3} \end{bmatrix}$

7.1 Individual Performance of the Algorithms

In this section, the performance of individual algorithms including QUEST, B-dot, MEKF with gyroscope measurements, MEKF with mathematical model, sliding-mode controller, and reaction wheel desaturation controller is examined. The performance of each algorithm is evaluated, and relevant comments and observations are provided.

7.1.1 QUEST

QUEST is a commonly used attitude determination algorithm in ADCS modes, either individually or as a measurement source for filters. Figure 7.1 shows the error in QUEST's ϕ angle estimation, as well as the angle between the two measurement sources used by QUEST: the sun sensor and magnetometer. This angle is determined by the dot product of the two measurements. When the measurements are parallel (angle of 0 or 180 degrees), the error in the QUEST algorithm increases significantly. This holds true for the other Euler angles θ and ψ as well.

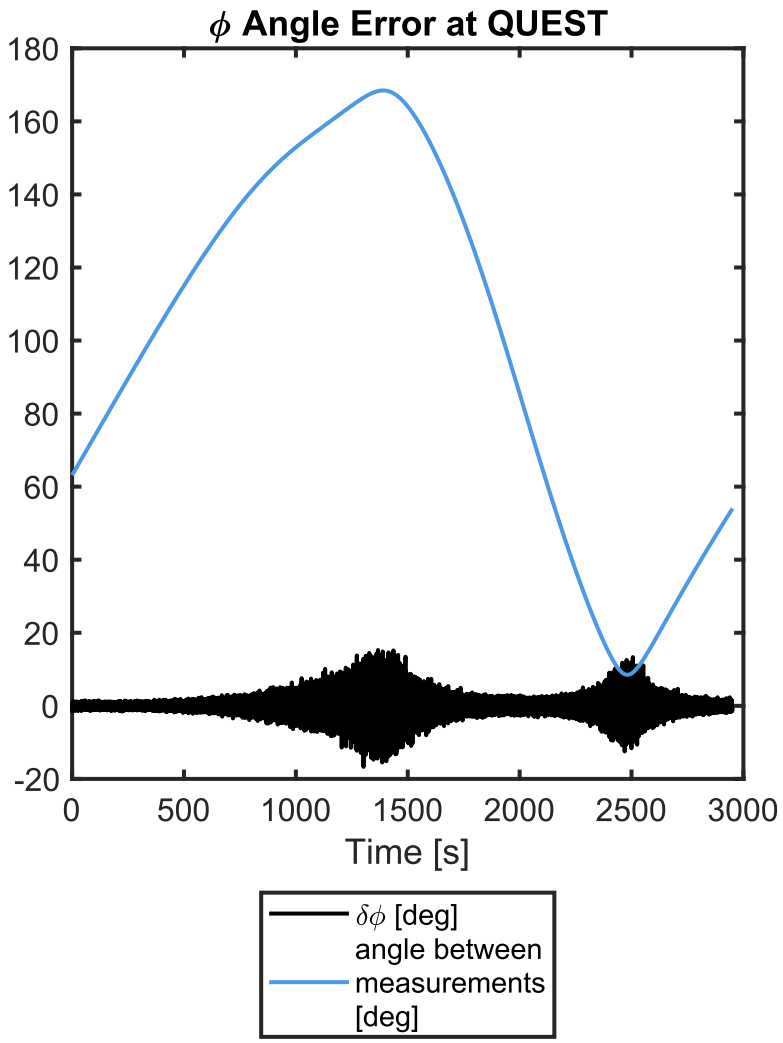


Figure 7.1: QUEST estimation error with angle between sun sensor measurement and magnetometer measurement

7.1.2 B-dot Controller

The B-dot controller is used to decrease the spacecraft's angular velocity. In Figure 7.2, the norm of the angular velocity is shown over time. The B-dot algorithm gradually reduces the magnitude of the angular velocity.

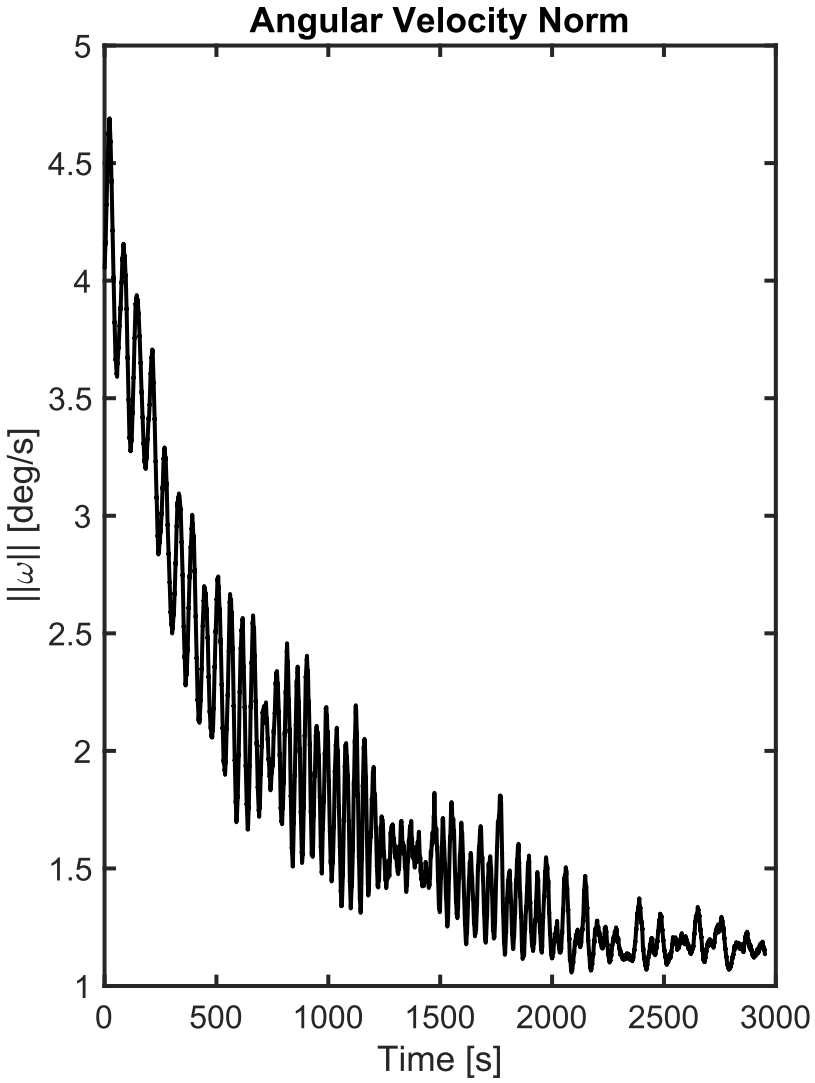


Figure 7.2: Angular velocity norm value reduction by using B-dot algorithm

7.1.3 MEKF

In the ADCS, two different MEKFs are utilized with two distinct measurement sources. In this subsection, the evaluation of filter and measurement source performance is conducted. The result of the MEKF with gyroscope measurement filters is considered more reliable than the MEKF with mathematical model. This is because it is generally difficult to model the dynamics accurately. There are various uncertainties in the system, either in the form of structured (parametric) uncertainties, which are modeled but

not accurately, or unstructured uncertainties, which are not modeled due to engineering judgment or being previously unknown [65]. Hence, the MEKF with gyroscope measurement is considered more reliable than the MEKF with mathematical model. Furthermore, the star tracker is a highly reliable sensor for quaternion measurement when compared to QUEST. This is because the main sources of the QUEST algorithm, namely magnetometer and sun sensor measurements, are not reliable. As shown in the BN constructed for the QUEST algorithm, the number of fault/error sources is higher than that of the star tracker (see Figure 5.2). To summarize, the filters and measurement source combination is presented below in decreasing order of confidence level.

1. MEKF with Gyroscope Measurement and Star Tracker
2. MEKF with Gyroscope Measurement and QUEST
3. MEKF wih Mathematical Model and Star Tracker
4. MEKF with Mathematical Model and QUEST

In the following section, the performance of the filter is evaluated based on the combination number. The combination number is determined according to the figure shown in Figure 5.4. First, to validate the functionality of the Kalman Filter, the errors of the states are checked to see if they fall within $\pm 3\sigma$. Then, the errors of the states are individually presented to provide a better understanding of the error level.

■ **Combination #1 (MEKF with Gyroscope Measurements and QUEST)**

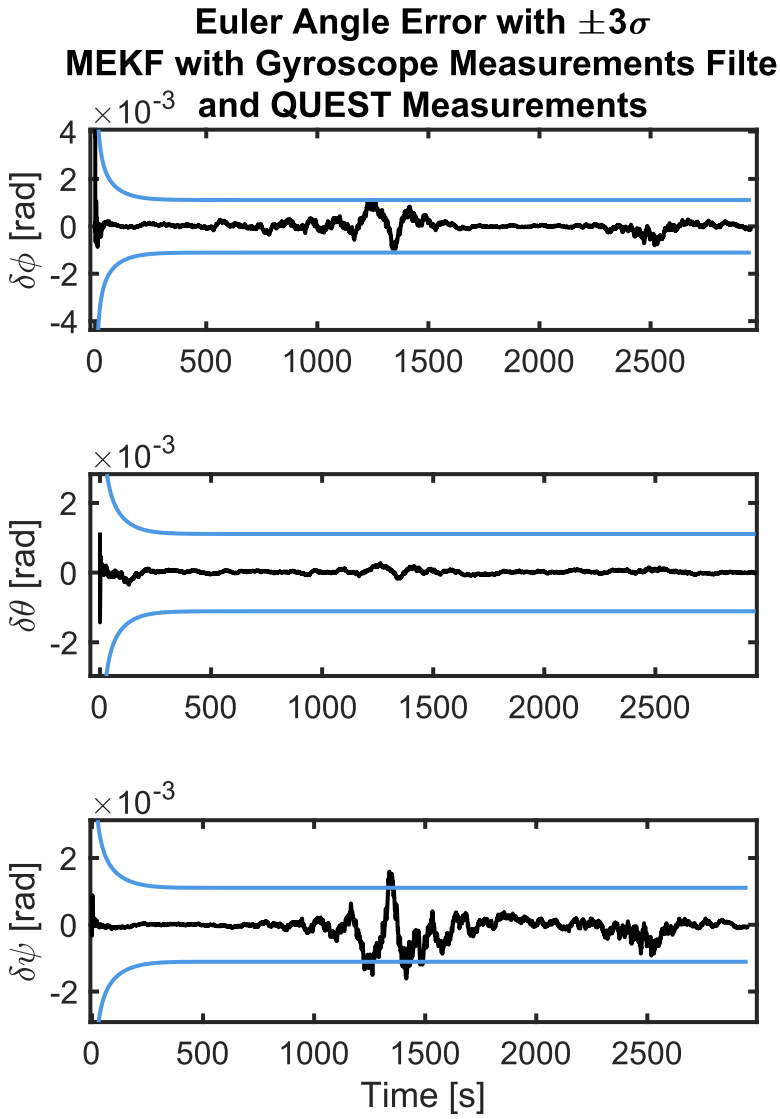


Figure 7.3: Combination #1 Euler angle errors with $\pm 3\sigma$

Figure 7.3 illustrates the Euler angle error of the MEKF with gyroscope measurement and QUEST filter, along with the theoretical $\pm 3\sigma$ value. The errors converge at around 160 [s], and the standard deviations converge to approximately 10^{-3} [rad].

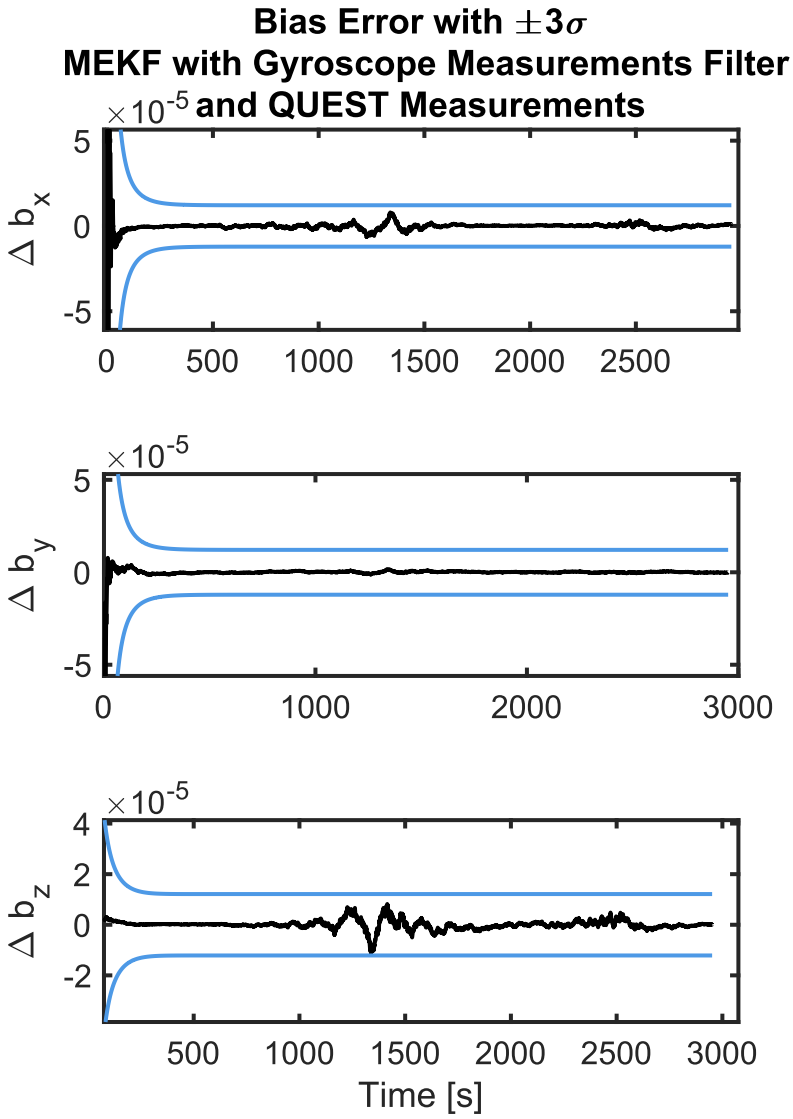


Figure 7.4: Combination #1 "other variable" errors with $\pm 3\sigma$

The convergence time for the gyroscope biases is longer than the convergence time for the Euler angles, at approximately 300 [s] (see Figure 7.4). The standard deviation of the bias at all three axes is close to 10^{-5} [rad/s], which is approximately 6×10^{-4} [deg/s].

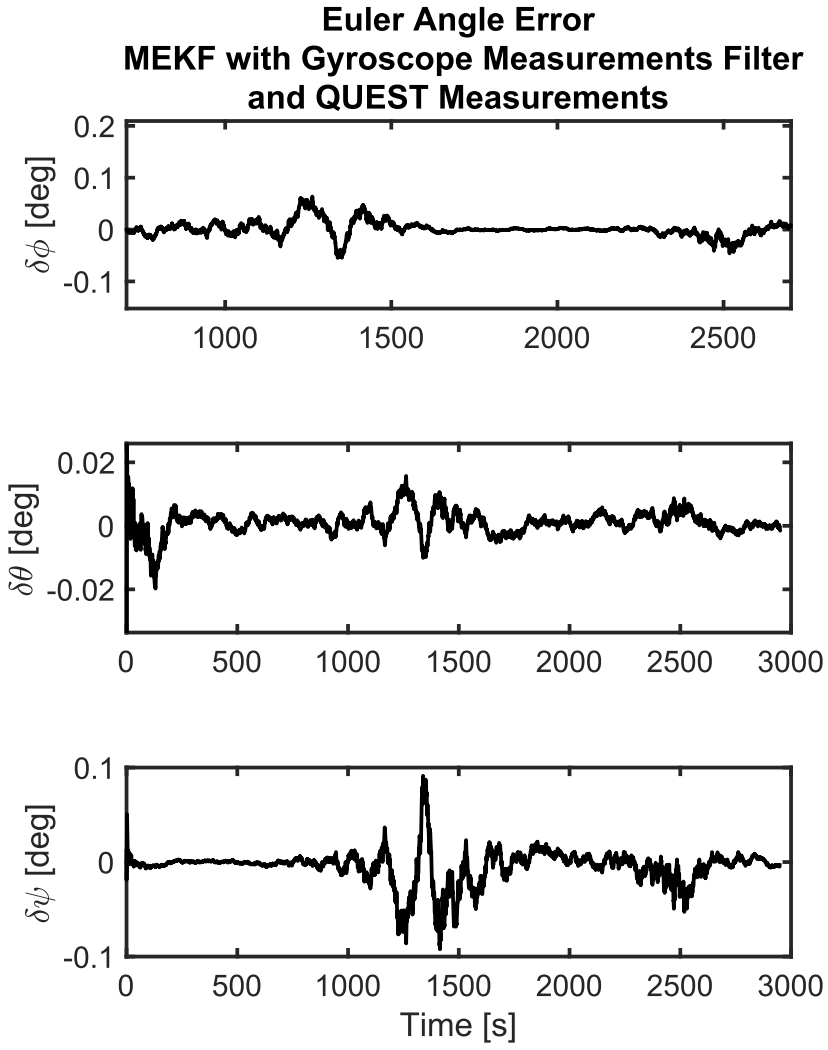


Figure 7.5: Combination #1 Euler angle errors

Figure 7.5 displays the Euler angle error magnitudes. As seen in Figure 7.3, the standard deviations are approximately 10^{-3} [rad] (≈ 0.05 [deg]), which is also evident from this figure. Furthermore, there is a notable increase in angle errors, particularly around 1500 [s] and 2500 [s]. This is because the quaternion estimations of the QUEST algorithm are not accurate during these time periods. This observation is supported by Figure 7.1, where around 1500 [s] and 3500 [s], the sun sensor and magnetometer measurements become collinear, indicating that the angle between the vectors is either 0 or π. Additionally, the error in the ϕ angle gradually becomes larger than the errors in the other Euler angles. This is because the magnetic field change is less noticeable during ψ rotation,

intuitively.

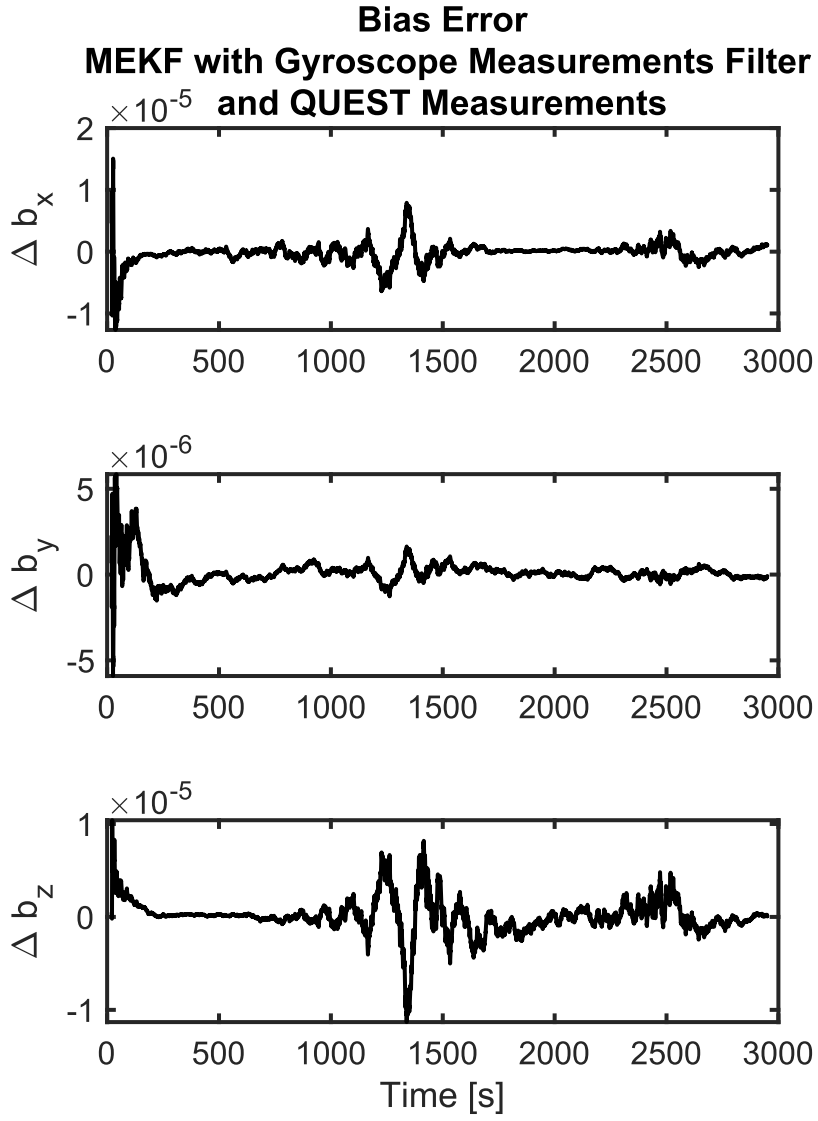


Figure 7.6: Combination #1 “other variable” errors

The gyroscope bias error is depicted in Figure 7.6. Similar to the observation made for the Euler angle error around 1500 and 2500 [s], the same trend can be observed for the bias error.

- **Combination #2 (MEKF with Gyroscope Measurements and Star Tracker)**

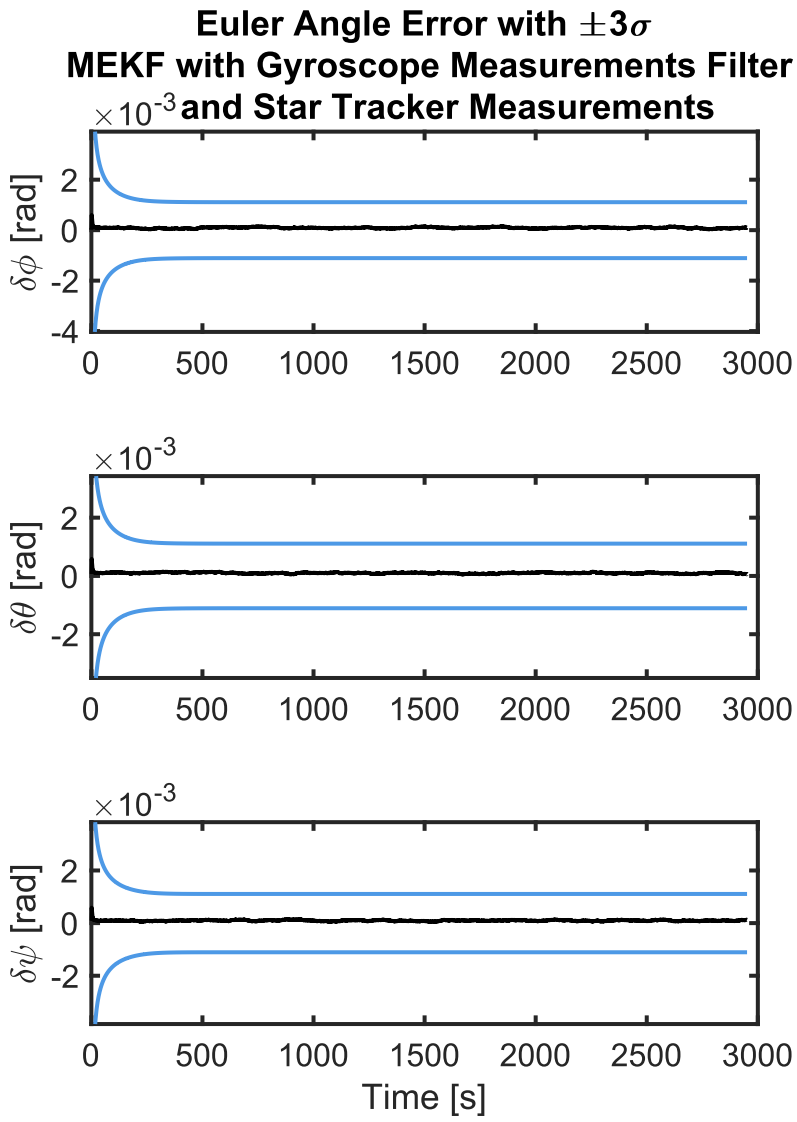


Figure 7.7: Combination #2 Euler angle errors with $\pm 3\sigma$

Figure 7.7 displays the Euler angle errors along with the theoretical bounds of the Kalman filter. The Euler angles converge around 80 [s]. However, the covariances converge again around 160 [s], similar to Combination #1 shown in Figure 7.3. This is because the same initial covariance, process noise, and measurement covariance values are used in both cases. However, as mentioned earlier, the star tracker provides a more accurate quaternion measurement compared to QUEST. Therefore, we expect the error convergence to occur much

earlier than the convergence of the standard deviations. Additionally, using the same covariance values yields a standard deviation of 10^{-3} [rad] for all three axes like in the Combination #1 case.

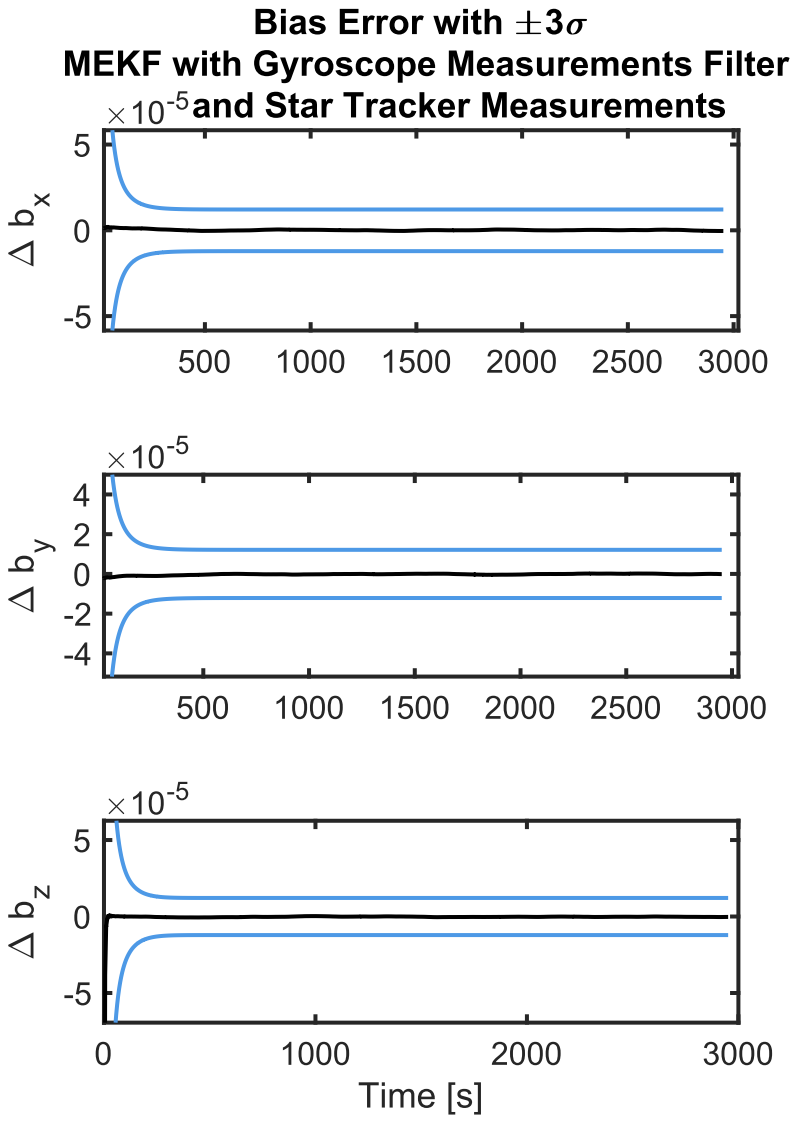


Figure 7.8: Combination #2 "other variable" errors with $\pm 3\sigma$

Figure 7.8 presents the gyroscope bias error. Once again, the error converges much faster than in the Combination #1 case. The same level of accuracy is observed, with a bias standard deviation of 10^{-5} [rad/s] $\approx 6 \times 10^{-4}$ [deg/s].

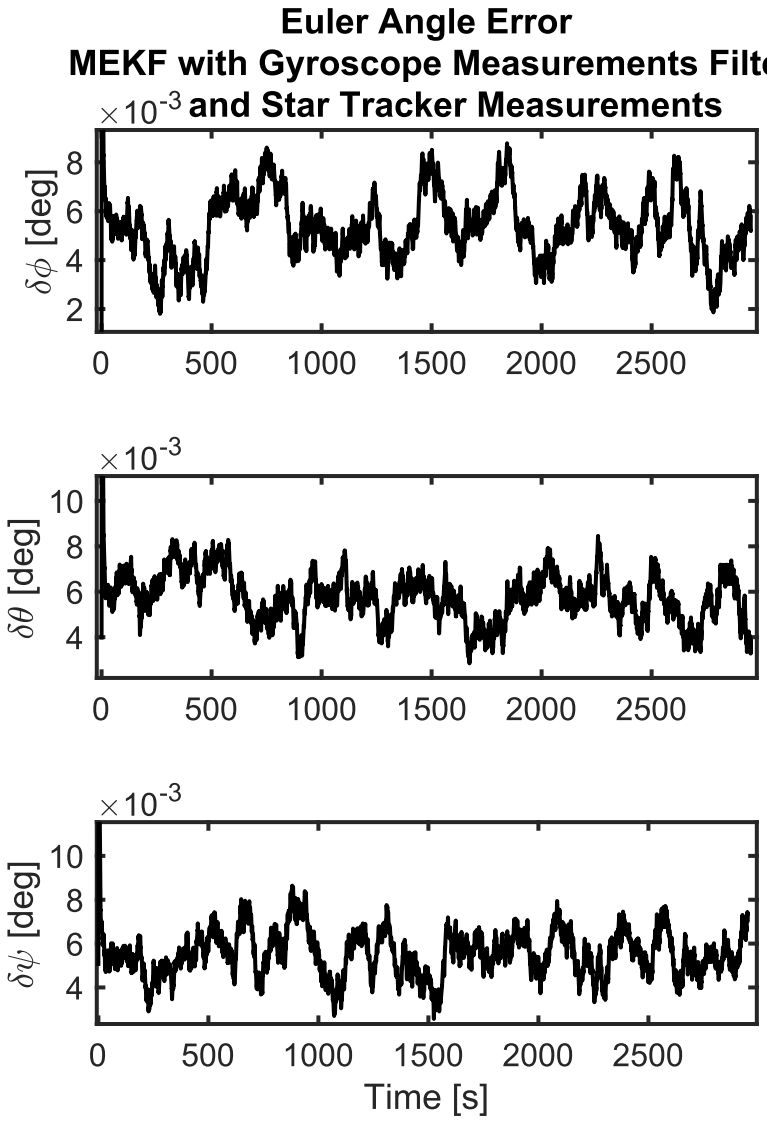


Figure 7.9: Combination #2 Euler angle errors

Figure 7.9 shows a very small Euler angle error at the level of $6{\times}10^{-3}$ [deg]. This is expected because Combination #2 is designed to provide the highest accuracy. Additionally, all Euler angles exhibit a similar error level. In Combination #1, the error in the ψ angle is higher than that in the other Euler angles. Intuitively, this can be attributed to the presence of the magnetic field in the orbit. These results support this idea.

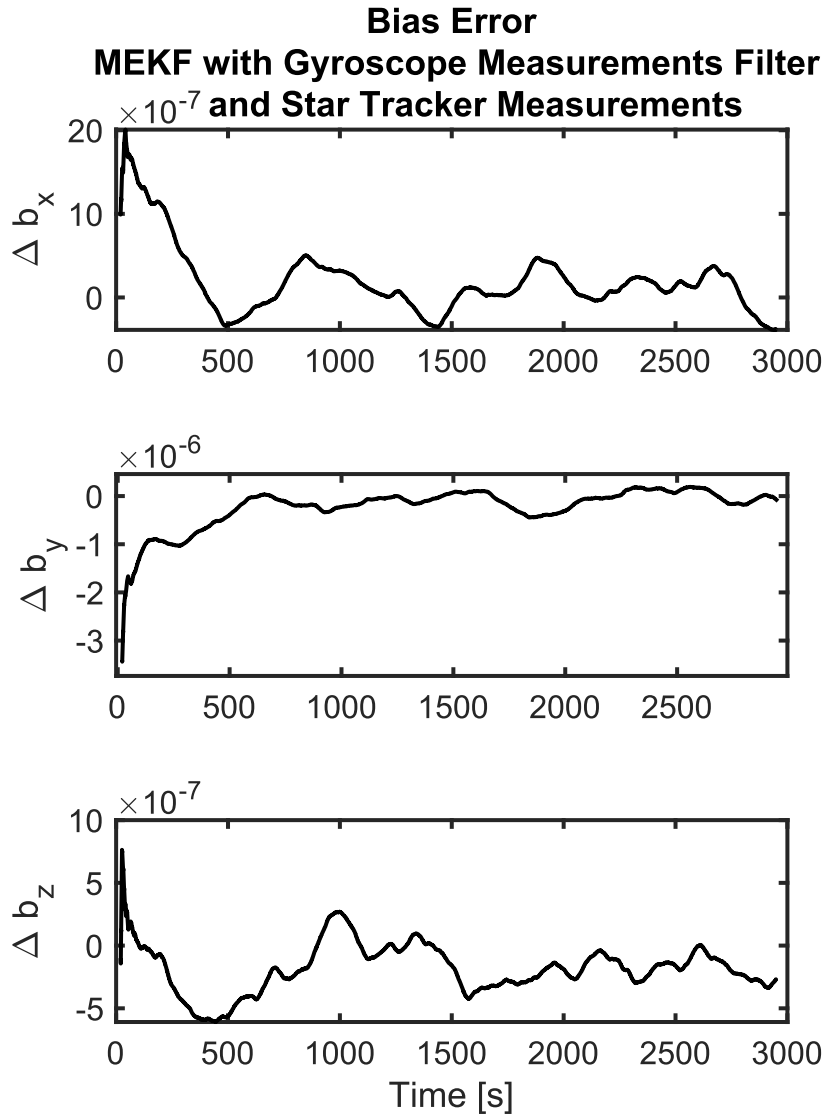


Figure 7.10: Combination #2 “other variable” errors

The gyroscope bias error in all three axes, as shown in Figure 7.10, is below 10^{-6} [rad/s] (approximately 6×10^{-5} [deg/s]).

- **Combination #3 (MEKF with Mathematical Model and QUEST)**

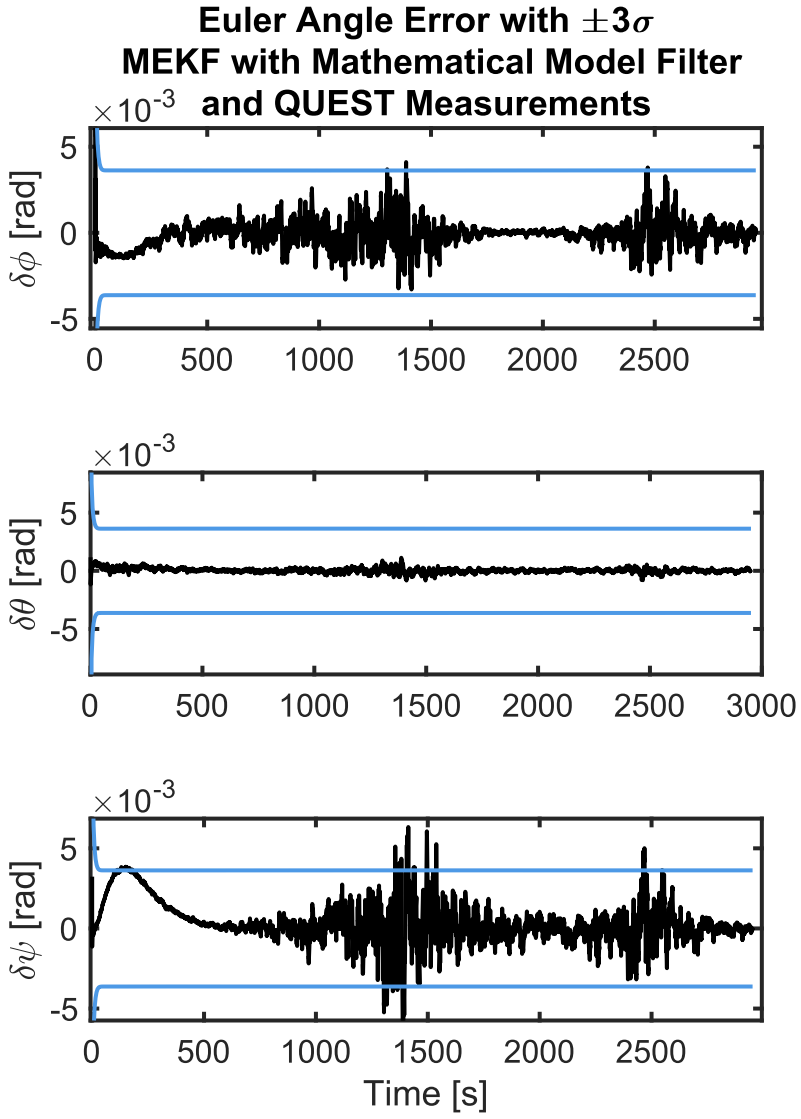


Figure 7.11: Combination #3 Euler angle errors with $\pm 3\sigma$

This combination exhibits a lower level of reliability. The convergence of errors occurs around 400 [s] for ϕ and θ, while for ψ it takes approximately 600 [s], Figure 7.11. Since QUEST is used as a measurement source, the convergence of ψ takes more time. Additionally, in the first 500 [s], a curved shape is observed in the angles. This is due to the filter operating with a sliding-mode controller, which has a greater impact during the initial 500 [s]. The convergence value of the standard deviation is approximately 3×10^{-3} [rad] for all three axes. This

value is higher than the Combination #1 and #2. Also, once again QUEST measurements clearly show low accuracy at around 1500 and 2500 [s].

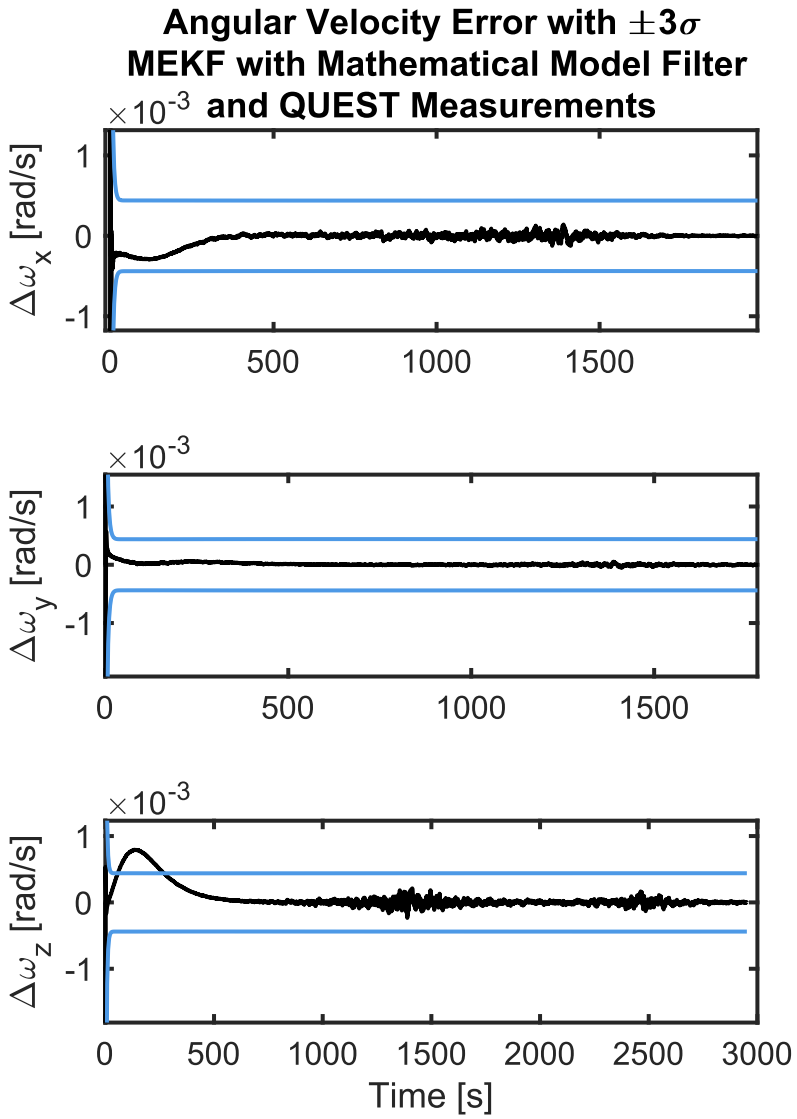


Figure 7.12: Combination #3 "other variable" errors with $\pm 3\sigma$

In Figure 7.12, the angular velocity converges for the x and y axes at approximately 400 [s]. However, for the z axis, the convergence occurs around 800 [s]. This can be due to the impact of QUEST specifically on the z-axis. Additionally, the bell-shaped curve observed in the first 500 [s] is more clear. This is likely due to the coupling between the estimator and sliding-mode controller. As this filter utilizes a mathematical model based on Euler's rotation, the mo-

ments generated by the sliding-mode controller can introduce inaccuracies. The convergence of the covariances is approximately 4×10^{-4} [rad/s] (approximately 0.02 [deg/s]).

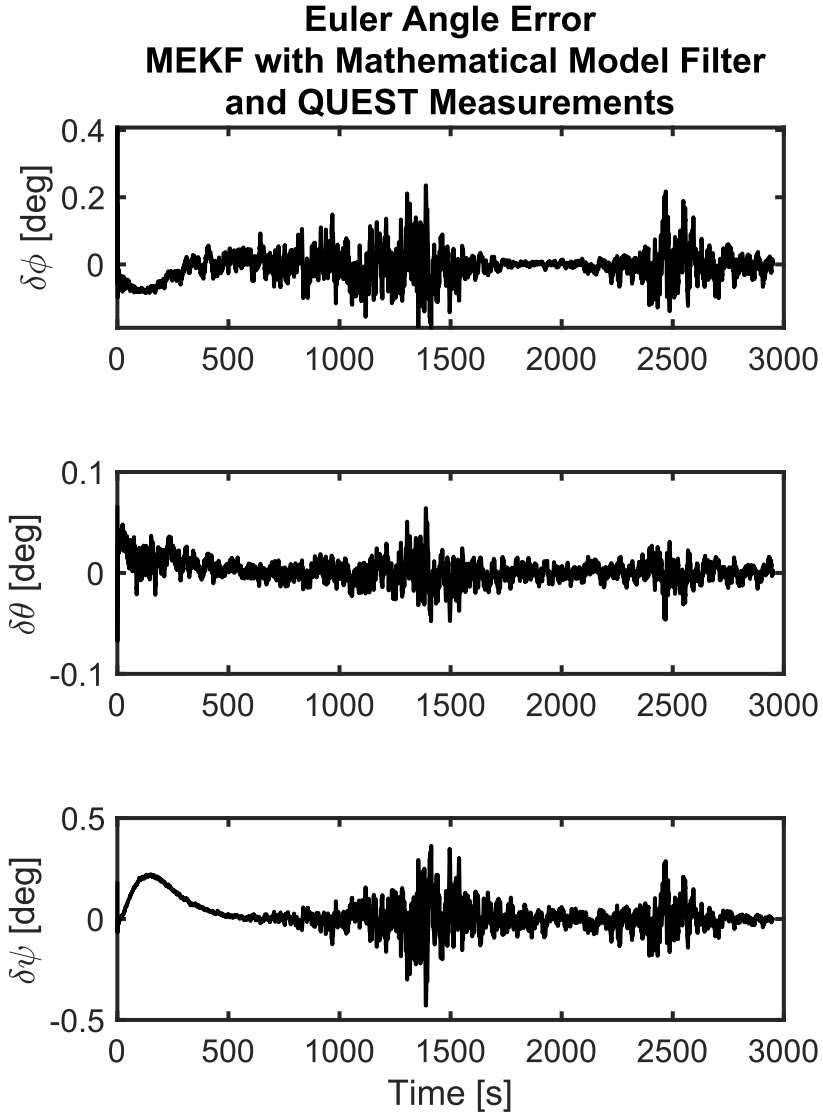


Figure 7.13: Combination #3 Euler angle errors

Figure 7.13 illustrates that the Euler angle error for ϕ is around 0.2 [deg/s], for θ it is less than 0.1 [deg/s], and for ψ it is approximately 0.5 [deg/s]. The error in the z-axis is larger compared to the other axes.

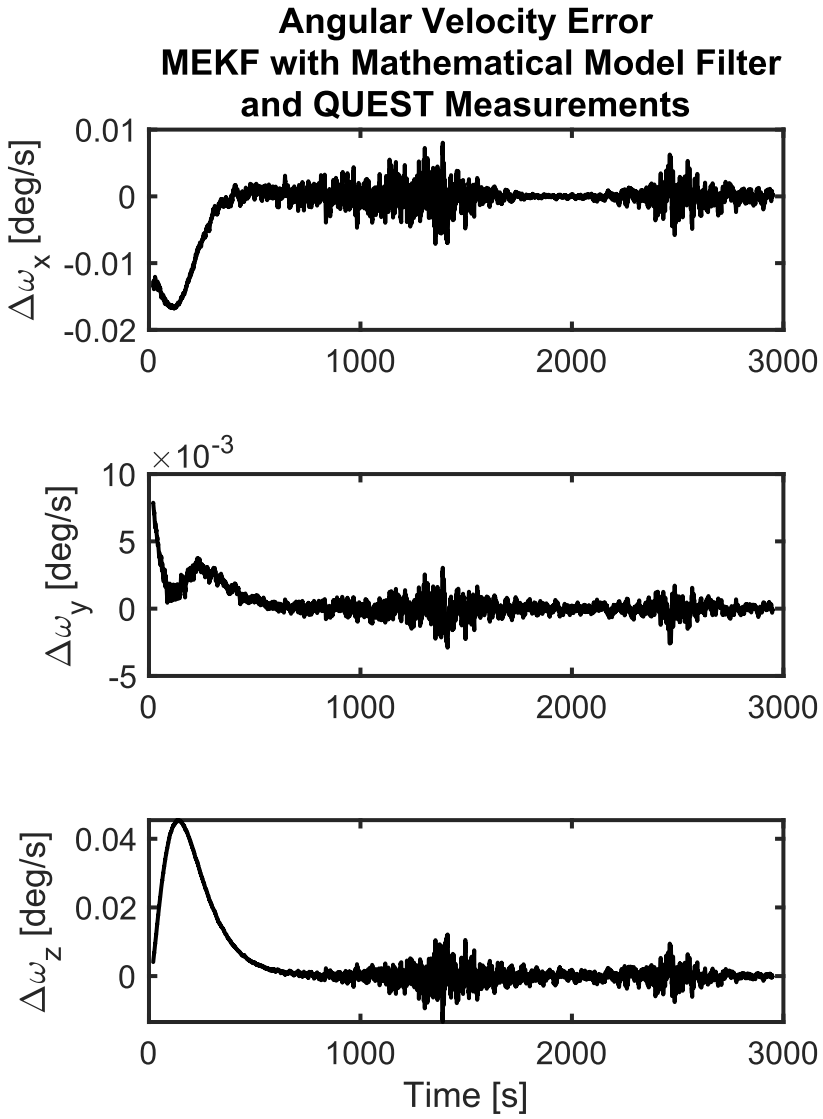


Figure 7.14: Combination #3 “other variable” errors

Once again, in Figure 7.14, the bell-shaped is most observable in the z-axis, and the error in ω_z is larger compared to the other axes.

- **Combination #4 (MEKF with Mathematical Model and Star Tracker)**

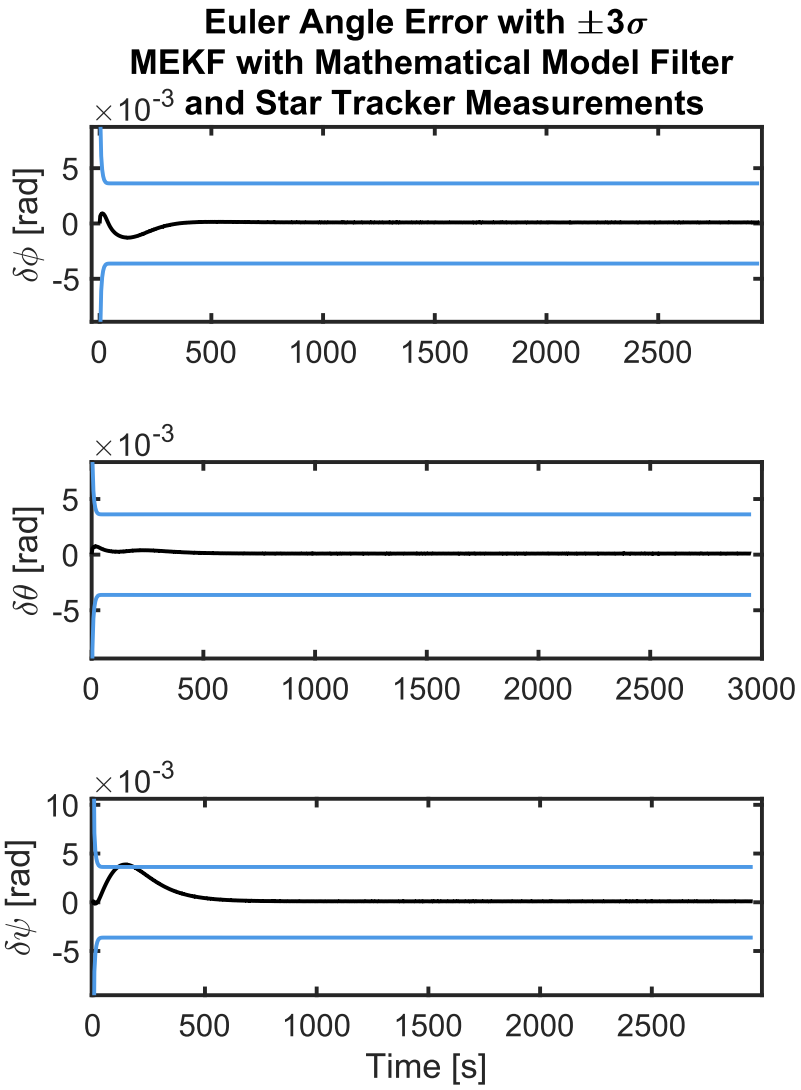


Figure 7.15: Combination #4 Euler angle errors with $\pm 3\sigma$

In Figure 7.15, the convergence of ϕ occurs around 400 [s], θ around 300 [s], and ψ around 500 [s]. The presence of bell-shaped curves in the first 500 [s] can be attributed to the sliding-mode controller operating with MEKF. The covariances converge to approximately 4×10^{-4} rad.

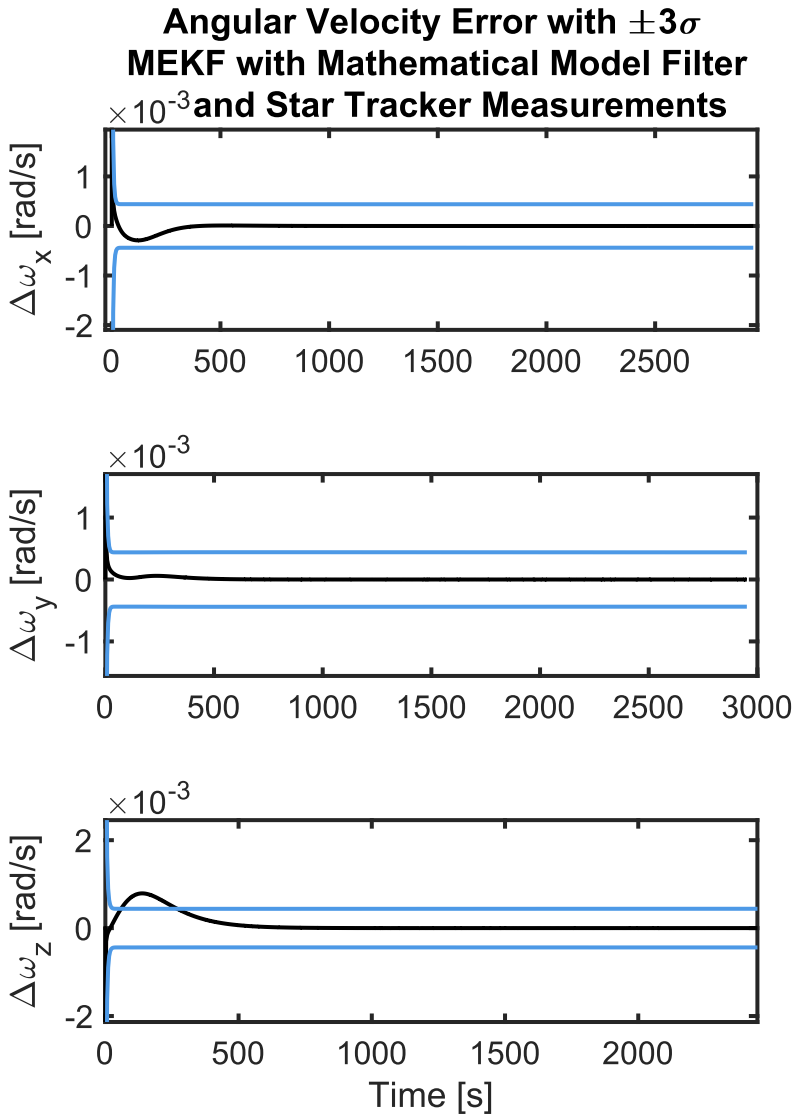


Figure 7.16: Combination #4 "other variable" errors with $\pm 3\sigma$

Figure 7.16 depicts the performance of angular velocity estimation for Combination #4. The convergence of ω_x occurs around 350 [s], while ω_z converges at approximately 600 [s]. On the other hand, ω_y converges rapidly. The covariances converge to approximately 4×10^{-4} [rad/s].

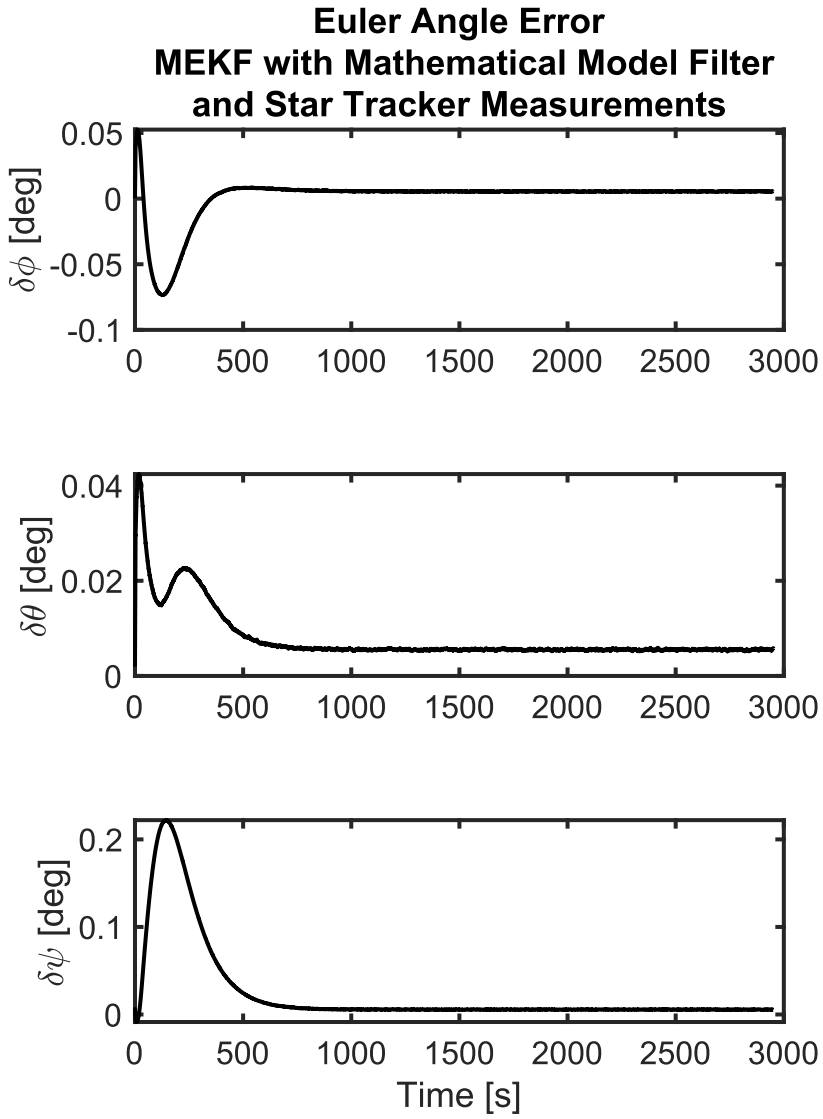


Figure 7.17: Combination #4 Euler angle errors

Figure 7.17 shows that Euler angle errors are small except the first 500 [s] bell-shaped behaviour. This an expected result because star tracker quaternion measurements are accurate.

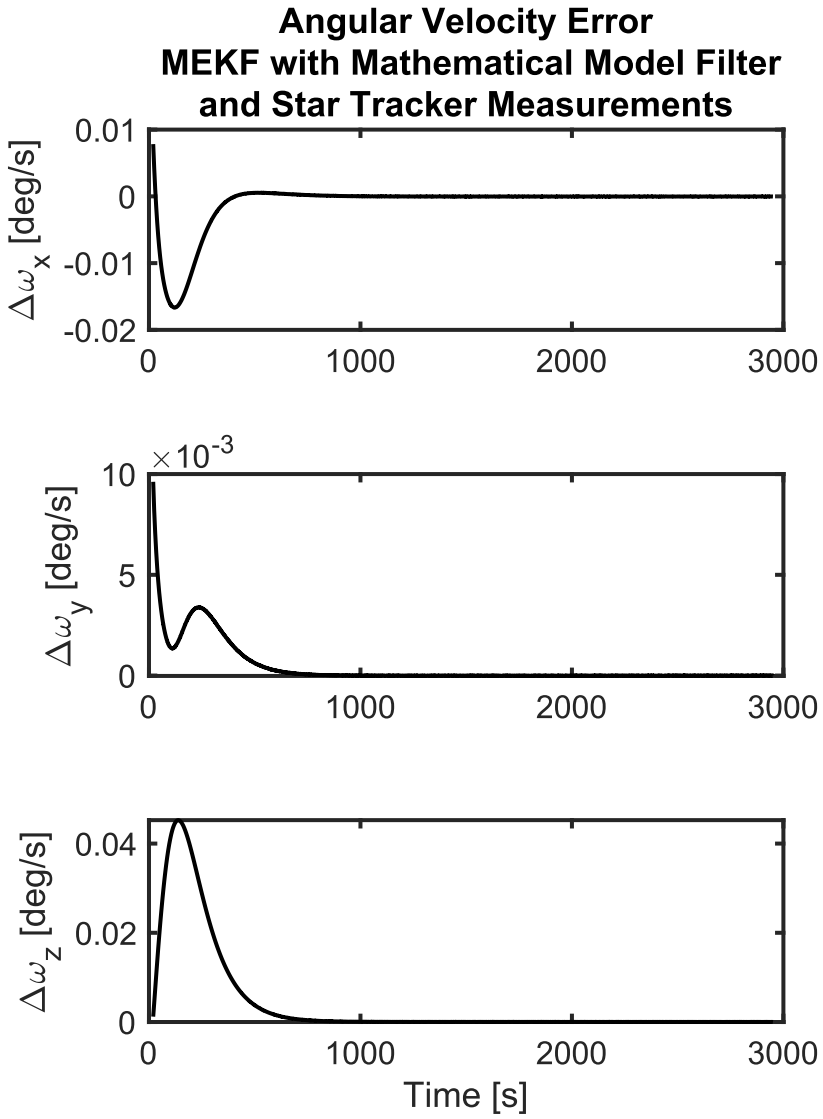


Figure 7.18: Combination #4 “other variable” errors

Figure 7.18 shows that angular velocity error is less than 0.05 [deg/s] for all three axis. Also, again due to sliding-mode high initial rates are observed at the first 500 [s].

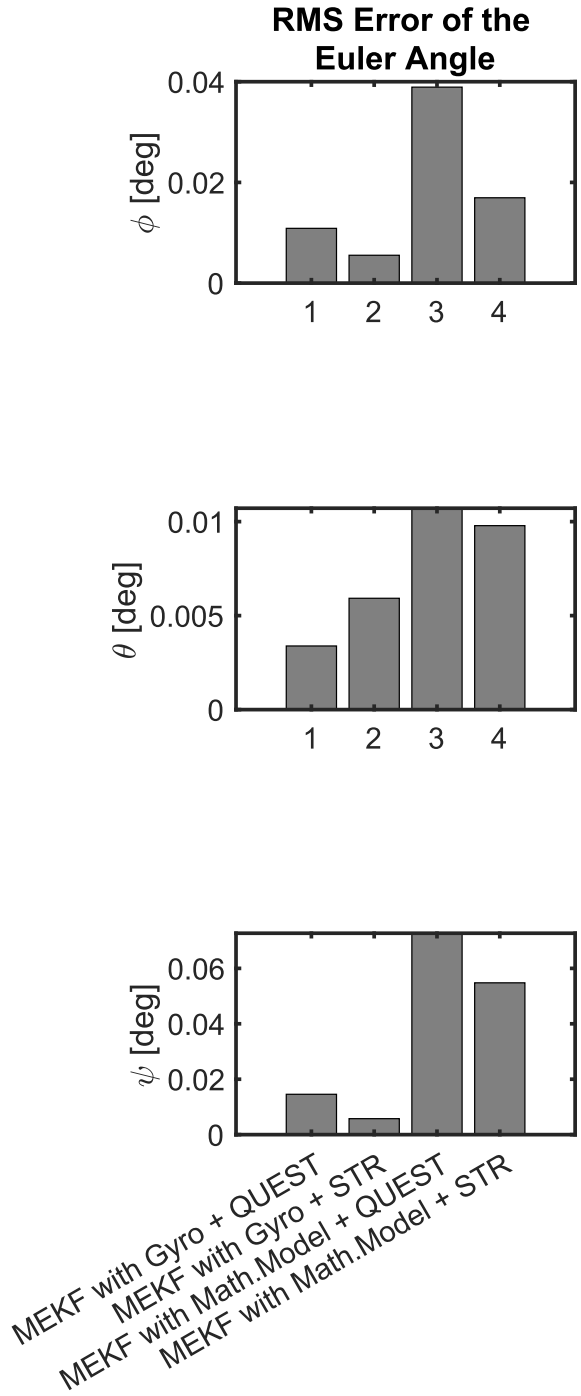


Figure 7.19: Root mean square error of the Euler angles with different filters and quaternion measurement source (STR represents the star tracker)

To sum up, MEKF with gyro measurements and star tracker is the most reliable combination. Then combinations can be ordered according to reliability: MEKF with

gyro measurements and QUEST, MEKF with mathematical model and star tracker, and MEKF with mathematical model and QUEST. To confirm this root mean square error of the Euler angle errors are taken, Figure 7.19. As shown in the Figure $\delta\phi$ and $\delta\psi$ is behave as expected, the same is not true for $\delta\theta$. However, error level of the $\delta\theta$ is small compare to other axis. In addition, QUEST algorithm results are only simulated by adding disturbance to sun sensor and magnetometer and in real world application magnetometer and sun sensor inaccuracies higher than simulated in this thesis. Moreover, these results are obtained under the assumption of the satellite performing nadir pointing. If the spacecraft's motion is altered, the outcomes can also change.

Lastly, to demonstrate the coupling effect between the MEKF with mathematical model filter and the sliding-mode controller, MEKF with mathematical model and QUEST were used with the B-dot controller. In Figure 7.20 and 7.21, it can be observed that there is no bell-shaped divergence when using this controller and estimator combination.

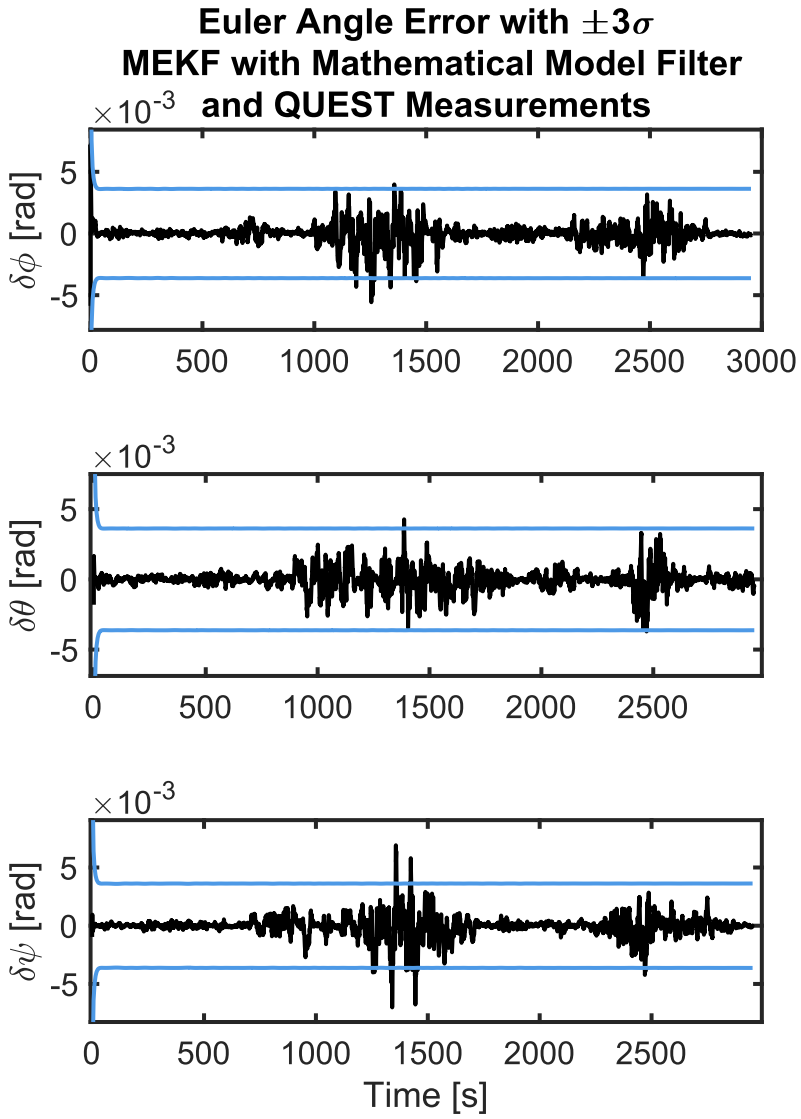


Figure 7.20: Combination #3 and B-dot controller Euler angle errors with $\pm 3\sigma$

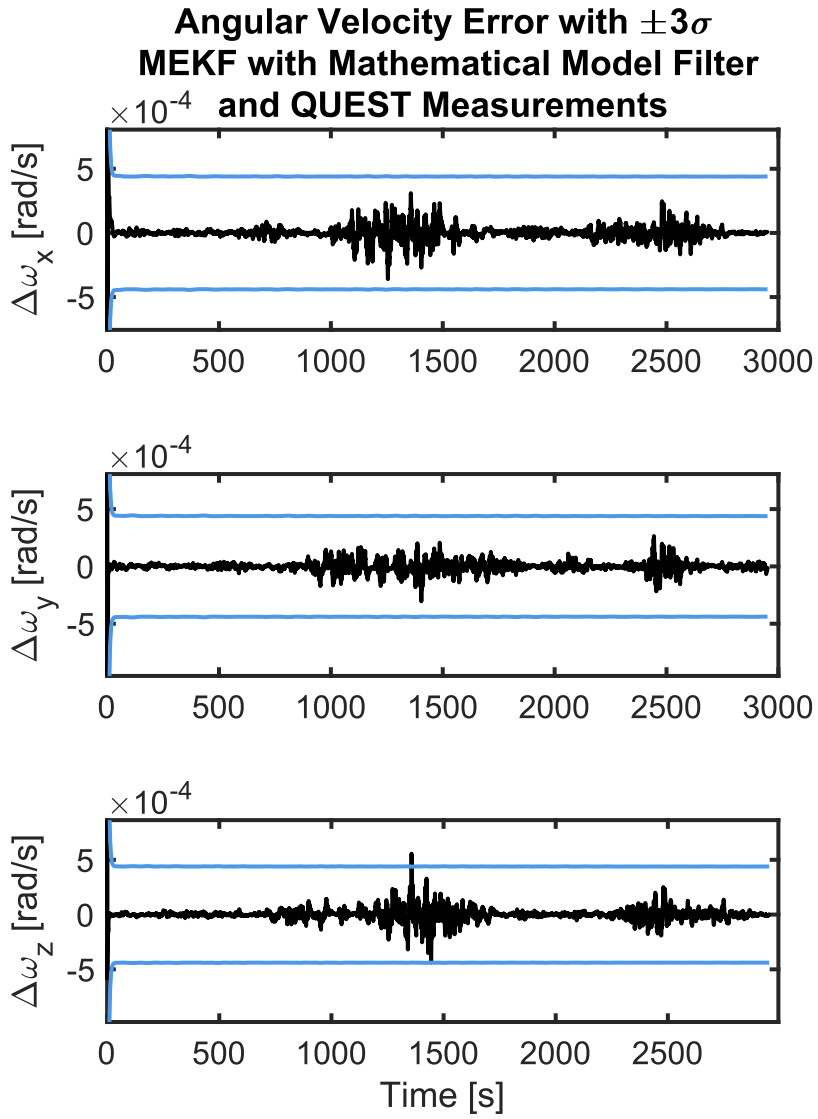


Figure 7.21: Combination #3 and B-dot controller "other variable" errors with $\pm 3\sigma$

7.1.4 Sliding Mode Controller

The sliding-mode controller is used in the main mission mode, which is mode 4. In Figure 7.22, various parameters related to the sliding-mode performance are plotted. The first parameter is the Euler angle, where the desired values are $\phi = 0$, $\theta = 0$, and $\psi = 0$. Over time, the Euler angles converge to their desired values. Additionally, since the main mission is Earth observation, the desired angular velocity ($\dot{\phi}$, $\dot{\theta}$, $\dot{\psi}$) is zero, therefore reaction wheel angular momentum are small. The difference between

the desired and measured angular velocity also approaches zero. Finally, the norm of the sliding surface is obtained. As expected, the magnitude of $\|\boldsymbol{s}\|$ is small and fluctuates.

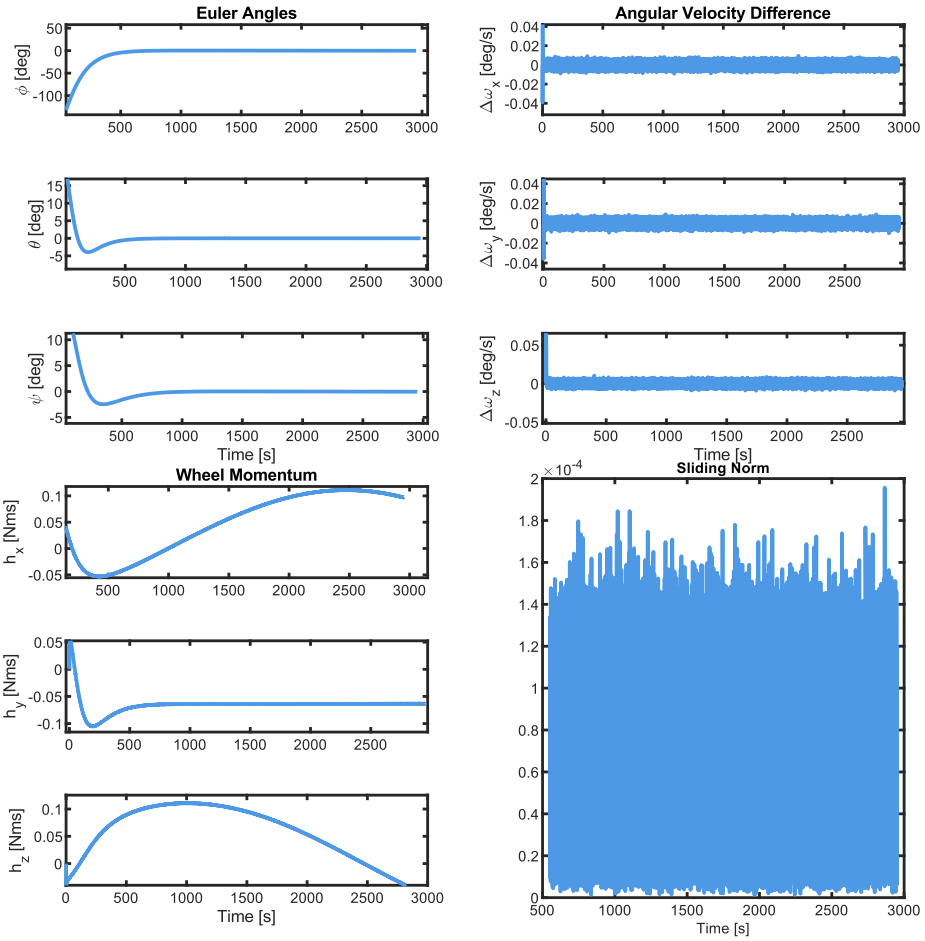


Figure 7.22: Sliding-mode controller performance

7.1.5 Reaction Wheel Desaturation Controller

This controller used when the reaction wheel saturated. As expected from the controller, reaction wheel momenta decreasing, Figure 7.23

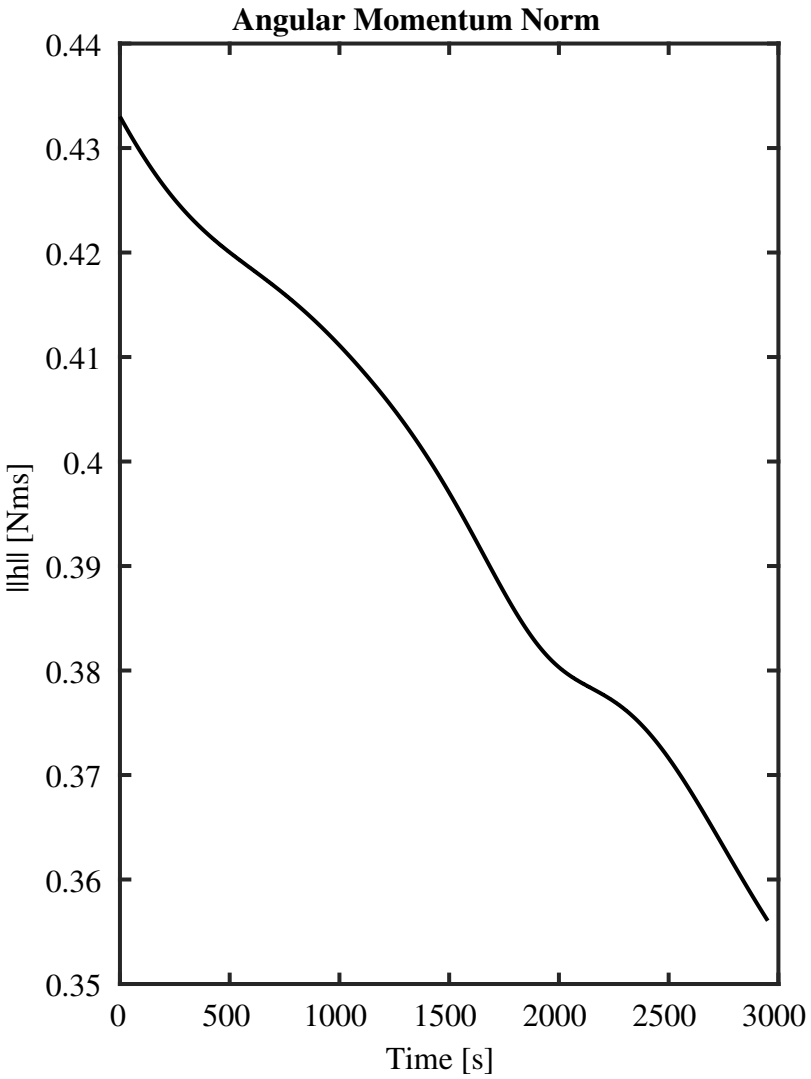


Figure 7.23: Angular momentum norm value reduction by using reaction wheel desaturation algorithm

7.2 Hybrid Performance of the Modes

This part focuses on examining the hybrid performance, which involves analyzing the mode transition and the interaction between different modes. Threshold values and Bellman optimization are both used in this analysis.

7.2.1 Hybrid Performance of the Modes with Thresholds

To demonstrate the hybrid performance, measurements of angular momentum of the wheel, angular velocity of the spacecraft, and Euler angle error of the estimation algorithms are taken for Earth-observing spacecraft. Note that all figures contain insets displaying specific values for mode 3. Mode 3 is designed as a transitional mode, resulting in a relatively shorter total time.

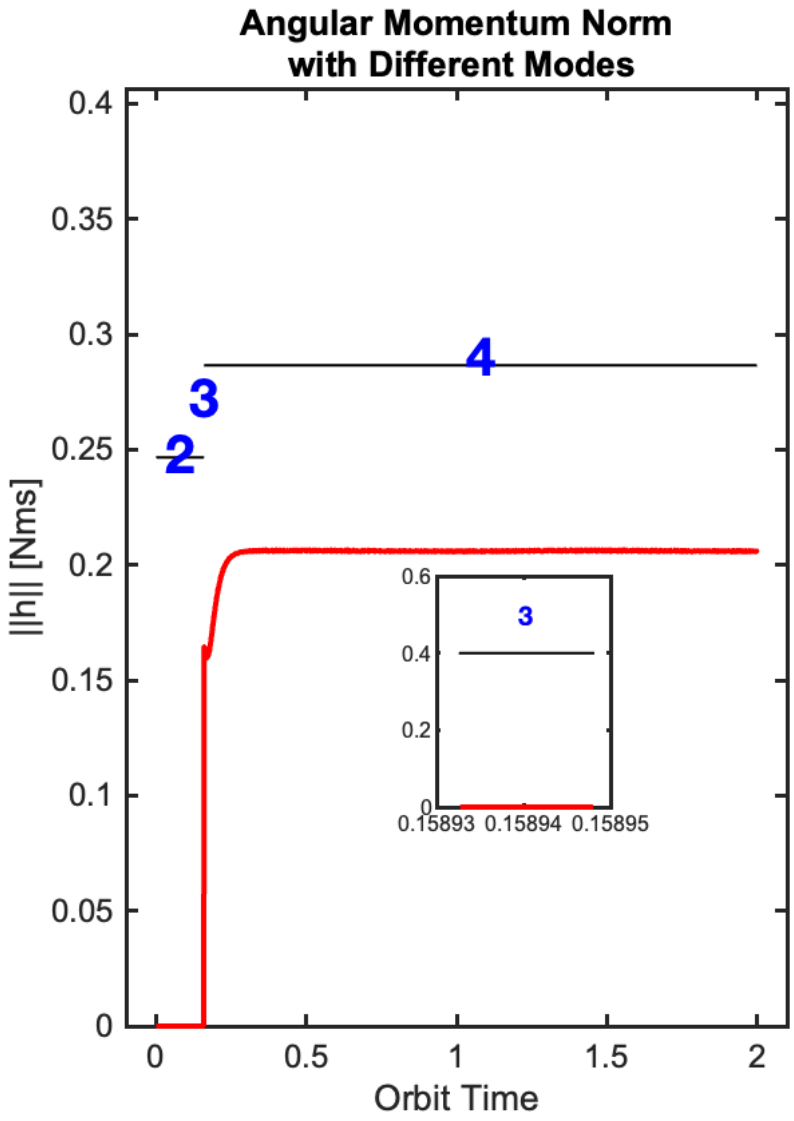


Figure 7.24: Mode transition with threshold, angular momentum

In Fig. 7.24, the wheel momentum is presented. The reaction wheel's angular momentum is initially set to zero at the start of the mission, and in modes 2 and 3, the magnetorquer is used as an actuator, resulting in a sharp rise in the angular momentum norm value as the mode switches to mode 4.

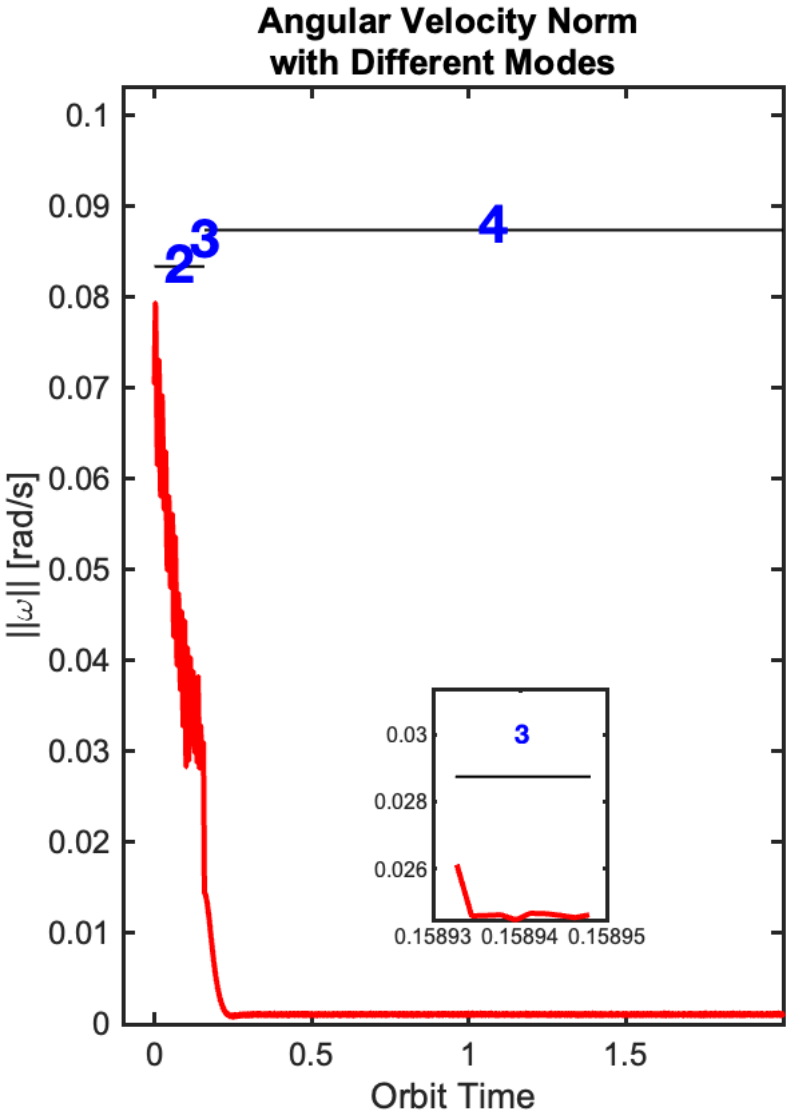


Figure 7.25: Mode Transition with threshold, angular velocity

Examining Fig. 7.25, it shows that the angular velocity of the spacecraft decreases during mode 2 and mode 3. In mode 4, this trends continues and reaching steady-state. This means that desired angular velocity for the sliding-mode controller is small.

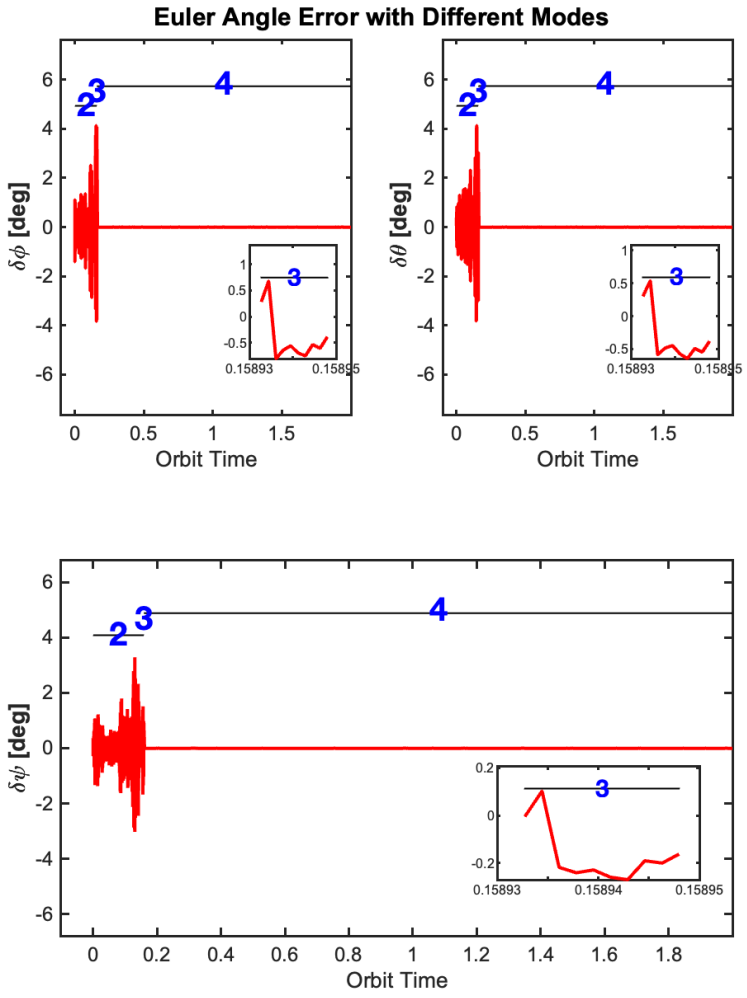


Figure 7.26: Mode Transition with Threshold, Euler Angles

Fig. 7.26 illustrates the angle error of the used algorithms, including QUEST, MEKF with gyroscope measurements, and MEKF with the mathematical model. The QUEST algorithm exhibits higher error compared to MEKF, resulting in a higher error value during mode 2. At the start of the transition from mode 2 to mode 3, the Euler angle error is around 4 [deg]. However, after applying MEKF in modes 3 and 4, the Euler angle error reduces to less than 0.01 [deg]. In fact, it is important to note that these values only provide relative attitude information between the QUEST algorithm and the MEKF. This is due to the fact that the simulation environment employed for this thesis does not model all types of disturbances.

Additionally, Fig. 7.24, 7.25, and 7.26 demonstrate that no angular momentum desaturation controller occurs in the hybrid performance. Another scenario was conducted by changing the desired angular velocity values. In the main mission of Earth-observing spacecraft, nadir pointing is typically employed. Consequently, the desired angular velocity values are not high and are determined according to Algorithm 8. In a new case, $\dot{\phi} = 0.03$ [rad/s], $\dot{\theta} = 0.02$ [rad/s], and $\dot{\psi} = 0.07$ [rad/s] were randomly defined, resulting in mode 5 being observed in the hybrid algorithm results. In the Earth-pointing scenario, $\dot{\phi} = \dot{\theta} = \dot{\psi} = 0$, and the desired Euler angles were also set to zero.

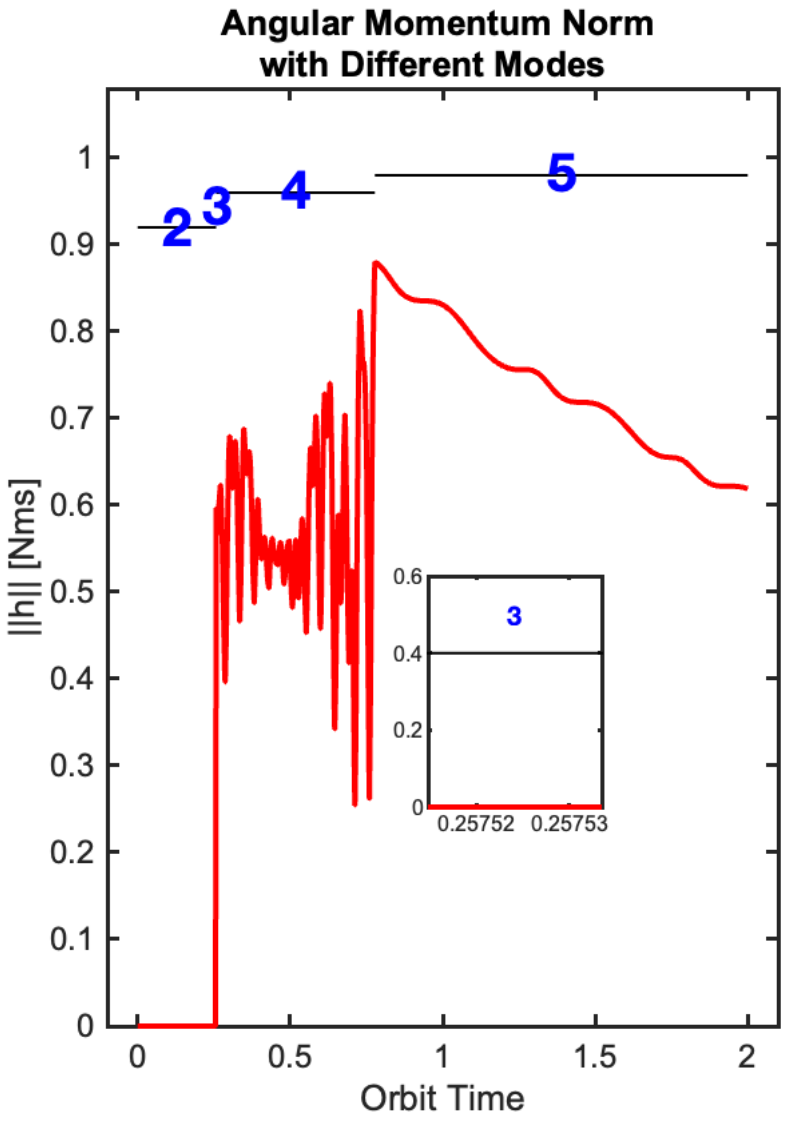


Figure 7.27: Mode Transition with Threshold, Angular Momentum

Figure 7.27 illustrates that when the wheel desaturation limit is exceeded, the ADCS switches. In mode 5, the spacecraft initiates the desaturation process.

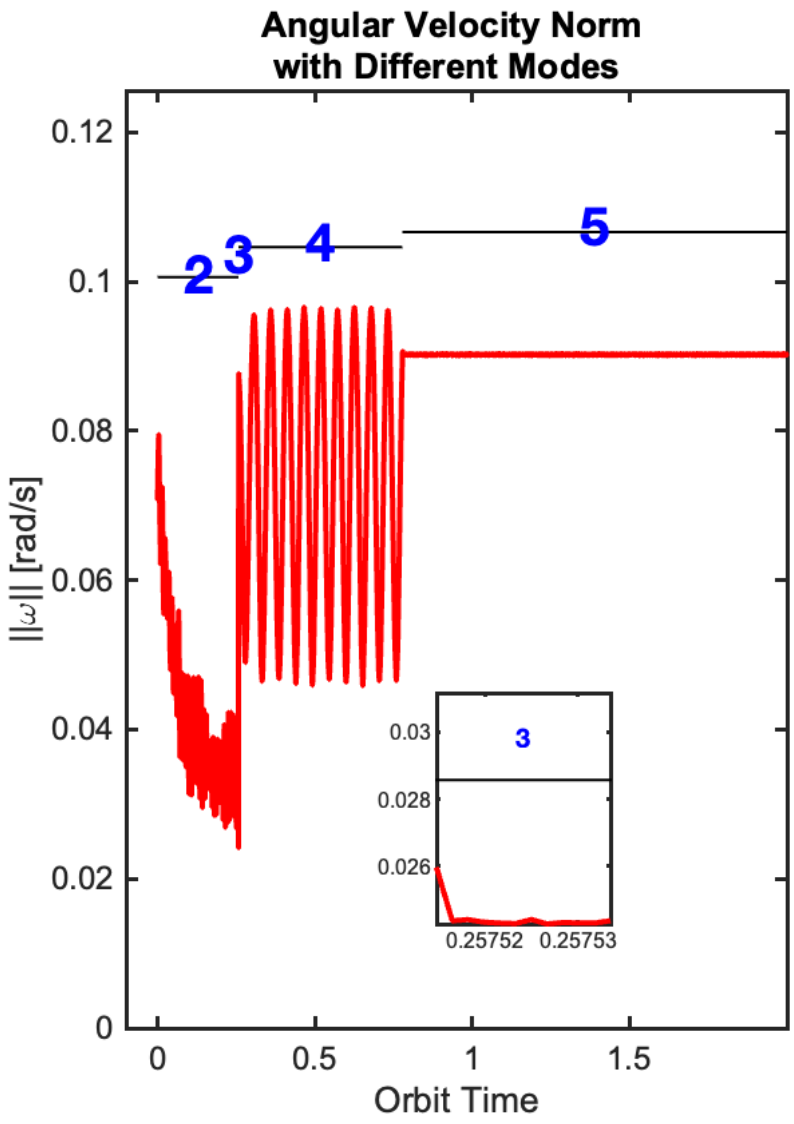


Figure 7.28: Mode Transition with Threshold, Angular Velocity

Fig. 7.28 displays the norm of the angular velocity across different modes. As expected, the spacecraft's angular velocity decreases in mode 2 and mode 3. However, after the ADCS switches to mode 4, the angular velocity experiences a sudden increase. This can be attributed to the commanded torque from the sliding-mode controller in order to achieve the desired attitude. Additionally, in mode 4, a periodic pattern in the angular velocity norm is observed. This is because, at given Euler rates,

the desired angular velocity shows periodic behavior, fluctuating between approximately 0.05 and 0.1 rad/s. Thus, this outcome is expected. In mode 5, the angular velocity remains constant. This is due to the total torque exerted on the spacecraft being zero, with the magnetorquer solely applying torque to desaturate the reaction wheel.

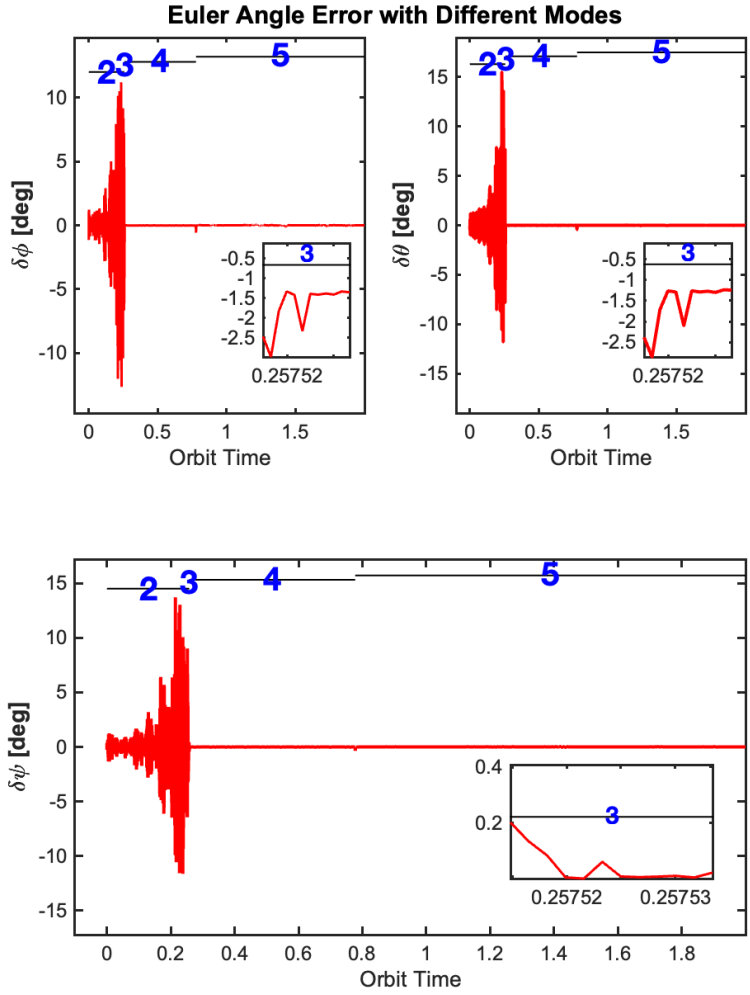


Figure 7.29: Mode Transition with Threshold, Euler Angles

In Fig. 7.29, the Euler angle errors in the QUEST algorithm reach 15 [deg]. This occurs around 1500 [s] (approximately 0.25 orbit time) when the angle between the measurement sources is π, Fig. 7.1. Consequently, the QUEST angle error increases significantly during this time in the figure. However, in modes 3, 4, and 5, the angle

errors remain low. There is only a small spike of approximately 0.6 degrees observed during the transition from mode 4 to mode 5.

7.2.2 Hybrid Performance of the Modes with Bellman Optimization

In this section, the same comment is applicable to Fig.7.24, 7.25, and 7.26. However, it is important to note that in the Bellman Optimization approach, mode 3 only is observed at a single point in time. The reason behind this is that mode 3 is designed to serve as a mid-phase between other modes. Unlike the mode transition based on threshold values, where a switch from Mode 3 to Mode 4 requires the specified condition to be met at least 10 times within a window of 20 samples, this is not the case for the Bellman Optimization method.

When analyzing the transition times from mode 3 to mode 4, a delay of 80 [s] is observed in the mode transition with Bellman optimization case. This transition occurs approximately at 0.17 times the spacecraft's orbital period. In comparison, for the threshold case, this transition takes place at approximately 0.16 times the orbital period. The spacecraft's orbital period is approximately 1.64 [hr].

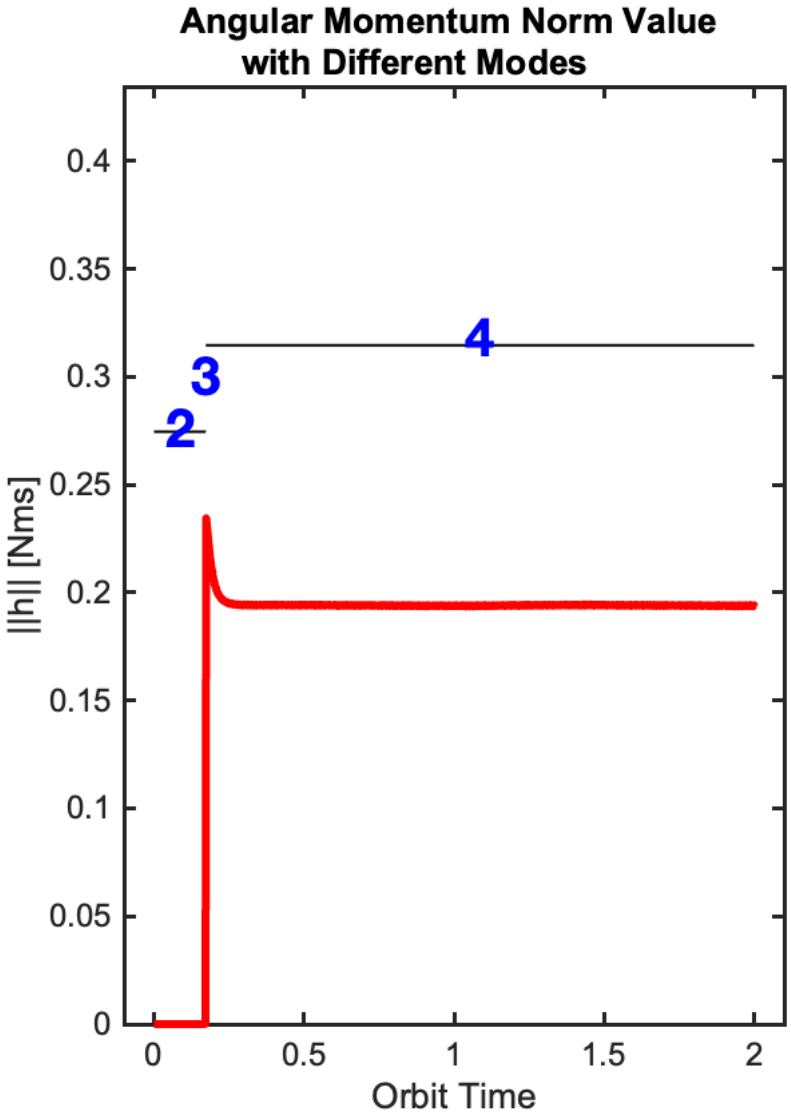


Figure 7.30: Mode Transition with Bellman Optimization, Angular Momentum

In Figure 7.30, the same behavior is observed as in the mode transition with threshold value shown in Figure 7.24. However, during the transition, when the reaction wheel starts to use wheel momentum, there is a sharp increase from 0 to 0.24 [Nms], which then converges to 0.2 [Nms]. On the other hand, in the mode transition with threshold case, a relatively slower increase from 0 to 0.16 [Nms] is observed, gradually reaching 0.2 [Nms]. Even though the same algorithms are used in the modes, the transition times are different. To understand the importance of the transition time, the Euler angle error graphs can be examined. When the mode transition occurs with Bellman

optimization, it is important to note that the Euler angle errors are approximately 6 [deg] (see Figure 7.32). However, when the mode transition takes place with threshold values, the Euler angle errors are approximately 4 [deg] (see Figure 7.26). This indicates that the Bellman optimization case requires the sliding-mode controller to utilize more controller power.

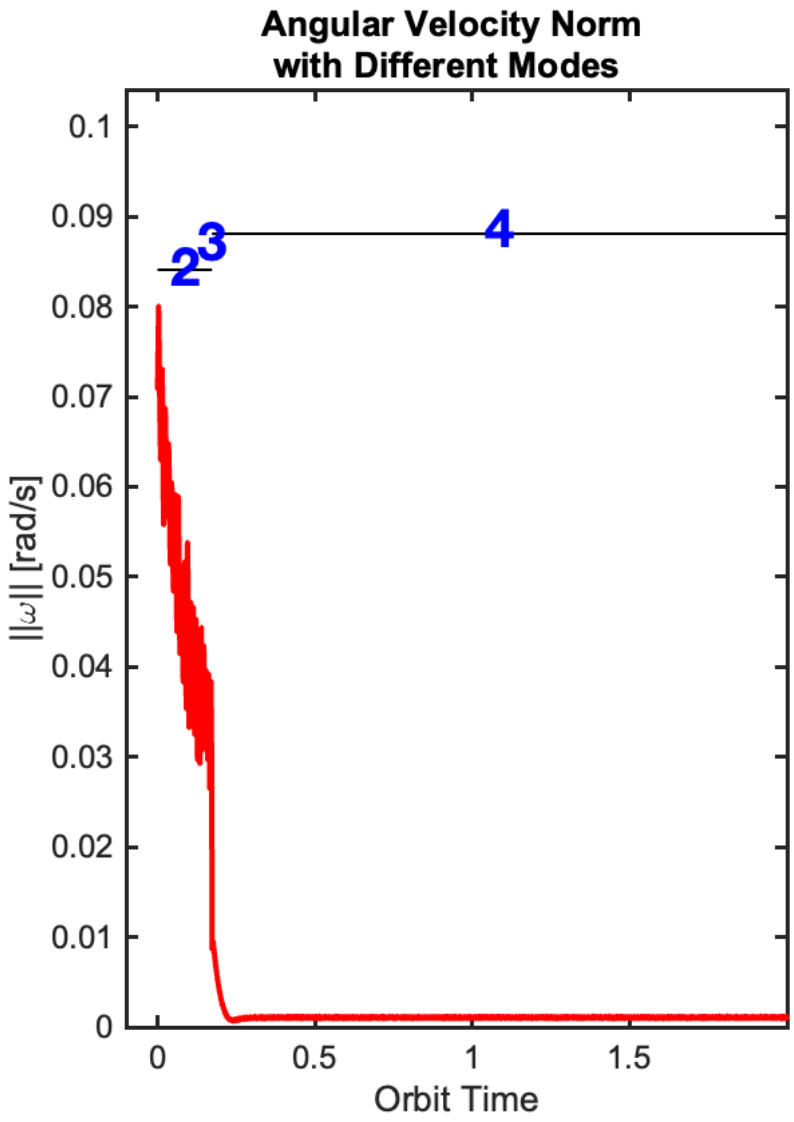


Figure 7.31: Mode Transition with Bellman Optimization, Angular Velocity

Fig. 7.25 and 7.31 exhibit a strong similarity. The same comments made for Fig. 7.25 can be applied to Fig. 7.31 as well.

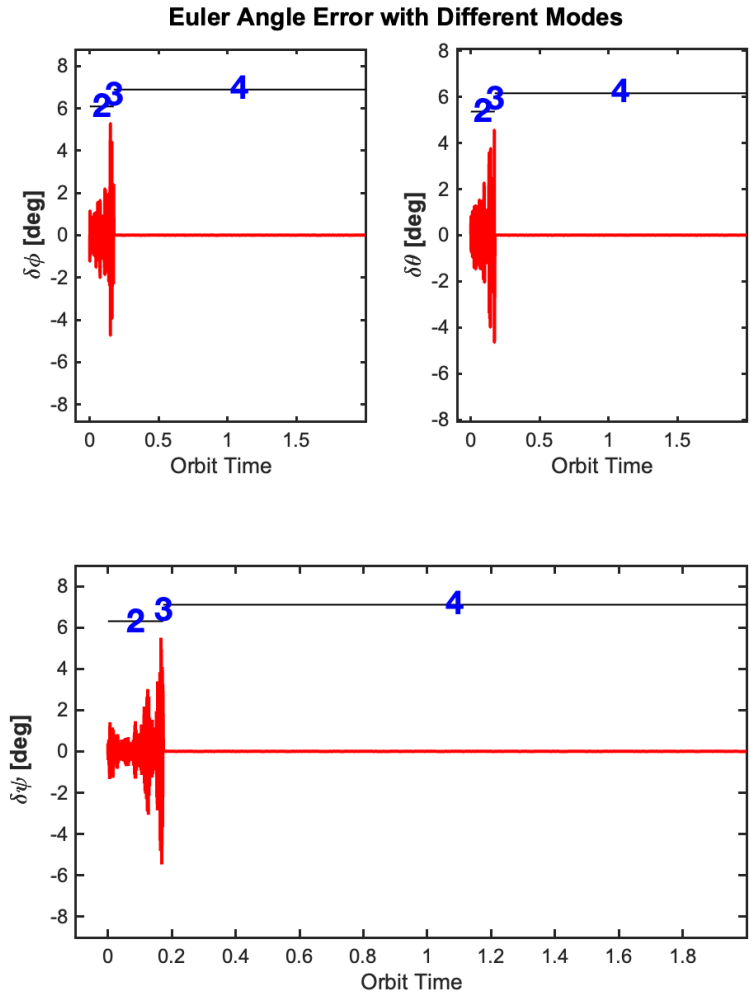


Figure 7.32: Mode Transition with Bellman Optimization, Euler Angles

Fig. 7.32 demonstrates similar trends observed in the mode transition with threshold values (Fig. 7.26). However, as mentioned earlier, due to different transition times, the Euler angle error at the start of the transition from mode 2 to mode 3 is approximately 6 [deg]. In the mode transition with threshold values, this value is 4 [deg] (Fig. 7.26). Nevertheless, when MEKF is employed in mode 4, this value converges back to less than 0.01 [deg].

7.3 Fault Detection and Isolation Algorithm Results

To assess how well the FDIR algorithm performs, three different tests have been designed. These tests involve observing nodes from four groups: QUEST, gyroscope, star tracker, and mathematical model nodes. However, it is important to note that in the gyroscope and mathematical model groups, there are nodes that are directly observable, namely "Gyro Fault" and "Mathematical Model Fault". These observable nodes serve as parents to the "MEKF Fault" node, which makes the fault source clear. Thus, testing is not performed on these groups. Furthermore, due to the absence of real data, no testing could be conducted to validate the overall BN network. Ideas for improving the results of the BN network are discussed in the Future Work section of the thesis.

As previously indicated, a value of "1" denotes the occurrence of a fault or error.

Test Case #1

Aim: In this test case, the main purpose is simulating fault at the “Measurement while Torque Rod Activating” node. For this purpose, magnetometer bias is increased.

Error Type: Magnetometer Bias Error - $bias\ of\ the\ magnetometer = 5000[nT]$

Observable Node:“Measurement while Torque Rod Activating” and “MEKF Fault” nodes

Active Graph (Filter): Graph#1

Expected Result: Fault/Error due to QUEST

$$P(QUEST\ Fault = 1 \mid MEKF\ Fault = 1,\ Meas.Torque\ Rod\ Activating = 1) = 0.9985$$

$$P(Gyro\ Fault = 1 \mid MEKF\ Fault = 1,\ Meas.Torque\ Rod\ Activating = 1) = 0.0832$$

Result = Fault/Error due to QUEST. Use Graph (Filter)#2 (Star Tracker + Gyro)

The fault that has been identified is a magnetometer bias error, which can be observed at the "Measurement while Torque Rod Activating" node. This indicates that the fault occurred during the execution of the QUEST algorithm. As anticipated, the results obtained from the BN are consistent with expectations. At this juncture, it may be appropriate to utilize Graph (Filter)#2.

Test Case #2

Aim: In this test case, the main purpose is simulating fault at the “Lens Blocked due to Bright Object” node. For this purpose, assuming that the star tracker can not give measurement at some time interval.

Error Type: Star Tracker does not give measurement - between [100,1000] [s]

Observable Node:“Lens Blocked due to Bright Object” and “MEKF Fault” nodes

Active Graph (Filter): Graph#2

Expected Result: Fault/Error due to Star Tracker

$$
\begin{aligned}
&P(Gyro\ Fault = 1 \mid MEKF\ Fault = 1,\\
&\quad Lens\ Blocked\ due\ to\ Bright\ Object = 1)\\
&\quad = 0.0897\\
&P(STR\ Fault = 1 \mid MEKF\ Fault = 1,\\
&\quad Lens\ Blocked\ due\ to\ Bright\ Object = 1)\\
&\quad = 0.9820
\end{aligned}
$$

Result = Fault/Error due to Star Tracker. Use Graph (Filter)#1 (QUEST + Gyro)

Once again, the BN is being employed to comprehend the cause of the fault or error, and the output obtained from the BN corroborates that the fault lies with the star tracker.

Test Case #3

Aim: In this test case, the main purpose is simulating fault at the "Orbit Determination Fault" node. Therefore, IGRF results are shifted 100 [s].

Error Type: IGRF Fault - 100 [s] shifting in the QUEST

Observable Node:"Orbit Determination Fault" and "MEKF Fault" nodes

Active Graph (Filter): Graph#1

Expected Result: Fault/Error due to QUEST

$$
\begin{aligned}
&P(QUEST\ Fault = 1 \mid MEKF\ Fault = 1,\\
&\quad IGRF\ Fault = 1)\\
&\quad = 0.9988\\
&P(Gyro\ Fault = 1 \mid MEKF\ Fault = 1,\\
&\quad IGRF\ Fault = 1)\\
&\quad = 0.0831
\end{aligned}
$$

Result = Fault/Error due to QUEST. Use Graph (Filter)#2 (Star Tracker + Gyro)

Given that the fault or error pertains to orbit determination, it is likely that the issue lies with the QUEST algorithm. The output of the BN also supports this conclusion.

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 Conclusion

The mission objectives have a significant impact on various aspects, including the ADCS modes, the accuracy of attitude pointing, and the choice of sensors and actuators for the ADCS system. In essence, these parameters are selected or designed based on the specific goals of a spacecraft's mission. Within the scope of this thesis, a 95 kg Earth-observation spacecraft's ADCS is developed. The main aim is to fulfill the general requirements common to Earth-observation satellites, hence specific details regarding attitude angles and rate accuracy are not explicitly defined. Instead, the modes and algorithms are tailored to meet overarching criteria. Furthermore, the spacecraft is placed in a sun-synchronous orbit at an altitude of 705 km.

Five distinct ADCS modes have been developed based on the spacecraft's mission timeline and specific mission requirements during different phases. These modes utilize three different estimation algorithms: QUEST, MEKF with gyroscope measurements, and MEKF with a mathematical model. Additionally, three distinct controllers are employed for these modes, namely the B-dot controller, sliding-mode controller, and reaction wheel desaturation controller.

The transition between different modes is achieved by using predetermined threshold values for both estimated and measured quantities. Moreover, the spacecraft's state-space is mathematically represented as a Markov Chain to do the transition in the different way. The modes are treated as inputs within the deterministic Markov Chains, and to use the optimal mode (input) Bellman Optimization techniques are employed. Utilizing Markov Chains for transitioning holds notable importance, particularly when

managing complex spacecraft tasks. This approach simplifies operations, being advantageous in scenarios where spacecraft are employed for complex missions within constellations.

This thesis employs BN as a FDI algorithm. The primary objective of using BN is to aid in selecting the appropriate filter type: either the MEKF with gyroscope measurements or the MEKF with a mathematical model. Additionally, BN assists in choosing the quaternion measurement source, which could be either QUEST or star tracker measurements. Expert knowledge is used in the construction process in the BN and conditional probability distributions. Furthermore, using BN for spacecraft ADCS is not common.

The study evaluates both individual and hybrid algorithm performance, encompassing mode transitions and corresponding outcomes. However, it's important to note that only gravity-gradient torque serves as the disturbance torque in this thesis, and no algorithm with hardware testing has been conducted. As a result, the pointing accuracies are not yet in their definitive state. Considering pointing accuracies in a relative manner, specifically in terms of mode-to-mode comparisons, is recommended.

8.2 Future Work

In this part, future work is outlined, highlighting potential areas for improvement and suggesting ideas to enhance the study.

- The probability distribution in the BN can be analyzed and improved in future study. Currently, the conditional probability distribution in the BN is constructed from expert knowledge. To enhance the study, a dataset can be utilized and a sensitivity analysis can be performed. Even though there isn't a dedicated dataset for studying MEKF faults, the literature can be used to create a MEKF failure dataset. This data can be used to evaluate the BN and enhance conditional probability distributions using learning. Additionally, conducting a sensitivity analysis will show the effect of different nodes on the overall results and help to improve the outcomes of the BN. Moreover, even though the focus of this thesis is on FDI utilizing the BN for MEKF faults, this strategy can be

extended to address faults in other modes of the spacecraft.

- In this study, TPMs were used to deterministically establish a mathematical model for mode transitions. For future research, it is suggested to utilize either a dataset or a simulation environment like Basilisk [2] to obtain a more precise mathematical model. In addition, Monte Carlo Tree search can be considered as an alternative to Bellman optimization, as it has been extensively studied in the literature [20, 26, 27].

- It is critical to include an extensive mathematical model that takes into account numerous disturbances in order to acquire more understanding of the algorithms' working. References such as [41] and [42] emphasize the significance of including disturbances such as gravity-gradient torque, solar radiation pressure, atmospheric drag, and torque due to magnetic residual moment. This study only simulates gravity-gradient torque, although other disturbances are significant. Modeling these disturbances would improve algorithm performance. It would also be beneficial to test the algorithms on other reliable simulators. A wide range of simulators used in the literature is provided in Reference [41], which can be used as an alternative platform for verifying and testing the algorithms. In general, using precise mathematical models that account for a wider range of disturbances and testing the algorithms on known simulators would increase their understanding and performance.

- When working with commercial off-the-shelf sensors, another critical point is the use of sensor calibration techniques. In a broad sense, these algorithms can be classified into two groups: algorithms for pre-processing and algorithms for correction in orbit. Prior to the satellite's launch, pre-processing algorithms are utilized to evaluate the sensors' performance under various conditions. On the other hand, in-orbit calibration algorithms are used to calibrate the sensors during the satellite's mission in space. In-orbit calibration is crucial because it is difficult to replicate the space environment on Earth. Reference [58, 63, 66] provide pre-processing calibration algorithms, while reference [63, 67] focus on in-orbit calibration methods. Furthermore, the impact of temperature changes on commercial off-the-shelf components during the satellite's orbit must be considered [63]. In the context of this thesis, no specific calibration algorithm has

been developed for addressing these issues. Thus, future study and development may include calibration methods and temperature impacts.

- Future study should involve test bench or hardware-in-the-loop experiments to improve the evaluation and improvement of ADCS algorithms [58]. Test bench or hardware-in-the-loop tests are crucial for fine-tuning algorithm parameters and enhancing the understanding of post-launch data analysis, as mentioned in reference [41].

- Accurately determining the spacecraft's position and velocity is another important aspect. Because the IGRF model is based on the spacecraft's position, the precision of orbit determination has a direct impact on the performance of estimate techniques. In the literature, two methods for orbit determination are commonly used: utilizing GPS receiver data or propagating the orbit using Two-Line Element (TLE) data [67]. However, the use of GPS receivers may be limited due to power consumption, and TLE methods are based on simplifying assumptions and the 2-body problem. A more sophisticated orbit determination technique is used in reference [16], including data from a GPS receiver, an Inertial Navigation System (INS), and TLE. It would be helpful for future studies to include an orbit determination part.

- Small spacecraft provide a suitable platform for testing new algorithms in space. Traditional algorithms like B-dot and momentum dumping rely on the perpendicularity of angular velocity and angular momentum to the magnetic field, which is not always applicable. New algorithms have been proposed in the literature to address this, as discussed in [8]. Additionally, using both magnetorquers and reaction wheels simultaneously has shown efficient performance [6]. New estimation algorithms have also been developed, with solar panels being utilized as sensors [8, 33]. Adaptive Kalman filters are important for achieving higher attitude accuracy in space applications. In particular, reference [30] emphasizes the need to adaptive process noise covariance matrix due to mathematical model disturbance changes along the orbit.

REFERENCES

[1] Introduction to Probability, Statistics and Random Processes, 2023.

[2] Welcome to Basilisk: an Astrodynamics Simulation Framework — Basilisk 2.2.0 documentation, June 2023. [Online; accessed 2. Jul. 2023].

[3] A. Adnane, Z. Ahmed Foitih, M. Si Mohammed, and A. Bellar. Real-time sensor fault detection and isolation for leo satellite attitude estimation through magnetometer data. *Advances in Space Research*, 61(4), 2018.

[4] A. Adnane, H. Jiang, M. Mohammed, A. Bellar, and Z. Foitih. Reliable kalman filtering for satellite attitude estimation under gyroscope partial failure. 2018.

[5] A. Ankan and A. Panda. pgmpy: Probabilistic graphical models using python. In *Proceedings of the 14th Python in Science Conference (SCIPY 2015)*. Citeseer, 2015.

[6] G. Avanzini, E. de Angelis, F. Giulietti, and N. Serrano. Attitude control of low earth orbit satellites by reaction wheels and magnetic torquers. *Acta Astronautica*, 160, 2019.

[7] G. Avanzini and F. Giulietti. Magnetic detumbling of a rigid spacecraft. *Journal of Guidance, Control, and Dynamics*, 35(4), 2012.

[8] D. Baldacchino, D. Debattista, R. Rotin, D. Cachia, M. Azzopardi, S. Fabri, and M. Bugeja. Review and feasibility of active attitude control and detumbling for the uombsat-1 pocketqube pico-satellite. 2017.

[9] P. Bangert, S. Busch, and K. Schilling. Performance characteristics of the uwe-3 miniature attitude determination and control system. volume 153, 2015.

[10] Y. Bar-Shalom, X. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. 2002.

[11] A. Barua and K. Khorasani. Hierarchical fault diagnosis and health monitoring in satellites formation flight. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 41(2), 2011.

[12] C. M. Bishop. *Pattern Recognition and Machine Learning*. Information science and statistics. Springer, 2006.

[13] A. Bolotskikh, R. Zharkikh, D. Ivanov, S. Lebedev, and S. Tkachev. Nanosatellite attitude motion determination algorithms study using laboratory facility. *Journal of Computer and Systems Sciences International*, 61(4), 2022.

[14] S. Bottone, D. Lee, M. O'Sullivan, and M. Spivack. Failure prediction and diagnosis for satellite monitoring systems using bayesian networks. 2008.

[15] S. Busch, P. Bangert, S. Dombrovski, and K. Schilling. Uwe-3, in-orbit performance and lessons learned of a modular and flexible satellite bus for future pico-satellite formations. *Acta Astronautica*, 117, 2015.

[16] A. Colagrossi and M. Lavagna. Fault tolerant attitude and orbit determination system for small satellite platforms. *Aerospace*, 9(2), 2022.

[17] J. Crassidis and J. Junkins. *Optimal estimation of dynamic systems*. 2004.

[18] A. Darwiche. *Modeling and reasoning with Bayesian networks*. 2009.

[19] P. B. Davenport. A vector approach to the algebra of rotations with applications. *NASA*, 1968.

[20] D. Eddy and M. Kochenderfer. Markov decision processes for multi-objective satellite task planning. 2020.

[21] H. Ehrpais, J. Kütt, I. Sünter, E. Kulu, A. Slavinskis, and M. Noorma. Nanosatellite spin-up using magnetic actuators: Estcube-1 flight results. *Acta Astronautica*, 128, 2016.

[22] B. Etkin. *Dynamics of Atmospheric Flight*. John Wiley Sons Inc, 1972.

[23] W. Feller. *An introduction to probability theory and its applications*. John Wiley.

[24] R. Fewster. Lecture notes in stochastic processes, 2015.

[25] C. Hajiyev and H. E. Soken. *Fault Tolerant Attitude Estimation for Small Satellites*. CRC Press, 1 edition, 2020.

[26] A. Herrmann and H. Schaub. Autonomous on-board planning for earth-orbiting spacecraft. volume 2022-March, 2022.

[27] A. P. Herrmann and H. Schaub. Monte carlo tree search methods for the earth-observing satellite scheduling problem. *Journal of Aerospace Information Systems*, 19(1), 2022.

[28] I. Hwang, S. Kim, Y. Kim, and C. Seah. A survey of fault detection, isolation, and reconfiguration methods. *IEEE Transactions on Control Systems Technology*, 18(3), 2010.

[29] T. Inamori, N. Sako, and S. Nakasuka. Attitude control system for the nano-astrometry satellite "nano-jasmine". *Aircraft Engineering and Aerospace Technology*, 83(4), 2011.

[30] D. Ivanov, D. Roldugin, S. Tkachev, M. Ovchinnikov, R. Zharkikh, A. Kudryavtsev, and M. Bychek. Attitude motion and sensor bias estimation onboard the siriussat-1 nanosatellite using magnetometer only. *Acta Astronautica*, 188, 2021.

[31] S. A. Jacklin. Small-satellite mission failure rates - nasa technical reports server (ntrs). *Small-Satellite Mission Failure Rates - NASA Technical Reports Server (NTRS)*, Apr 2019.

[32] V. Kattakuri. Failures in spacecraft systems: An analysis from the perspective of decision making, Aug 2019.

[33] O.-J. Kim, H. Shim, S. Yu, Y. Bae, C. Kee, H. Kim, J. Lee, J. Han, S. Han, and Y. Choi. In-orbit results and attitude analysis of the snuglite cube-satellite. *Applied Sciences (Switzerland)*, 10(7), 2020.

[34] D. Koller and N. Friedman. *Probabilistic graphical models : principles and techniques*. Adaptive computation and machine learning. MIT Press, 2009.

[35] P. R. Kumar and P. Varaiya. *Stochastic Systems: Estimation, Identification, and Adaptive Control*. Society for Industrial and Applied Mathematics, 2016.

[36] D. Y. Lee, H. Park, M. Romano, and J. Cutler. Development and experimental validation of a multi-algorithmic hybrid attitude determination and control system for a small satellite. *Aerospace Science and Technology*, 78, 2018.

[37] E. Mabrouk. What are SmallSats and CubeSats? *NASA*, 2015.

[38] D. J. C. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2002.

[39] F. Markley. Unit quaternion from rotation matrix. *Journal of Guidance, Control, and Dynamics*, 31(2), 2008.

[40] F. Markley and J. Crassidis. *Fundamentals of spacecraft attitude determination and control*. 2014.

[41] C. Menegaldo, F. De O. Fialho, E. Janot-Pacheco, F. Pait, and V. Lapeyrère. Picsat's enduring legacy. probing the flight of a small astronomical satellite. *Publications of the Astronomical Society of the Pacific*, 134(1033), 2022.

[42] K. Miyata and J. Van Der Ha. Attitude control by magnetic torquer. volume 134, 2009.

[43] M. Mughal, J. Praks, R. Vainio, P. Janhunen, J. Envall, A. Näsilä, P. Oleynik, P. Niemelä, S. Nyman, A. Slavinskis, J. Gieseler, N. Jovanovic, B. Riwanto, P. Toivanen, H. Leppinen, T. Tikka, A. Punkkinen, R. Punkkinen, H.-P. Hedman, J.-O. Lill, and J. Slotte. Aalto-1, multi-payload cubesat: In-orbit results and lessons learned. *Acta Astronautica*, 187, 2021.

[44] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. Adaptive Computation and Machine Learning. The MIT Press, 2012.

[45] Y. Nakajima, N. Murakami, T. Ohtani, Y. Nakamura, K. Hirako, and K. Inoue. Sds-4 attitude control system: In-flight results of three axis attitude control for small satellites. volume 19, 2013.

[46] S. Nakasuka, K. Miyata, Y. Tsuruda, Y. Aoyanagi, and T. Matsumoto. Discussions on attitude determination and control system for micro/nano/pico-satellites considering survivability based on hodoyoshi-3 and 4 experiences. *Acta Astronautica*, 145, 2018.

[47] A. Nash and A. Austin. Fundamental problems in smallsat concept development. 2021.

[48] I. Ofodile, J. Kütt, J. Kivastik, M. Kaspar Nigol, A. Parelo, E. Ilbis, H. Ehrpais, and A. Slavinskis. Estcube-2 attitude determination and control: Step towards interplanetary cubesats. 2019.

[49] B. Ould Bouamama, G. Biswas, R. Loureiro, and R. Merzouki. Graphical methods for diagnosis of dynamic systems: Review. *Annual Reviews in Control*, 38(2), 2014.

[50] M. K. Ozgoren. Lecture notes in advanced dynamics. Middle East Technical University, Department of Mechanical Engineering, 2020-2021.

[51] D. Paulson, D. Jackson, and C. Brown. Spars algorithms and simulation results. *Proceedings of the Symposium on Spacecraft Attitude Determination*, 1, 1969.

[52] H. Pishro-Nik. *Introduction to Probability, Statistics, and Random Processes*. Kappa Research, LLC, 2014.

[53] J. Praks, M. Mughal, R. Vainio, P. Janhunen, J. Envall, P. Oleynik, A. Näsilä, H. Leppinen, P. Niemelä, A. Slavinskis, J. Gieseler, P. Toivanen, T. Tikka, T. Peltola, A. Bosser, G. Schwarzkopf, N. Jovanovic, B. Riwanto, A. Kestilä, A. Punkkinen, R. Punkkinen, H.-P. Hedman, T. Säntti, J.-O. Lill, J. Slotte, H. Kettunen, and A. Virtanen. Aalto-1, multi-payload cubesat: Design, integration and launch. *Acta Astronautica*, 187, 2021.

[54] K. Przytula, F. Hagen, and K. Yung. Bayesian networks for satellite payload testing. volume 3812, 1999.

[55] M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. Wiley Series in Probability and Statistics. Wiley-Interscience, 1 edition, 1994.

[56] M. K. Quadrino. Testing the attitude determination and control of a CubeSat with hardware-in-the-loop, 2014.

[57] F. Reichel, P. Bangert, S. Busch, K. Ravandoor, and K. Schilling. The attitude determination and control system of the picosatellite uwe-3? volume 19, 2013.

[58] Y. Sato, S. Fujita, T. Kuwahara, Y. Shibuya, and K. Kamachi. Design, implementation and in-orbit demonstration of attitude and orbit control system for micro-satellite ale-2. 2021.

[59] H. Schaub. Kinematics: Describing the motions of spacecraft, 2023. Accessed on April 9, 2023.

[60] D. Shah. Algorithms for inference. Massachusetts Institute of Technology, 2014.

[61] S. Shi, B. Yuan, K. Zhao, Z. You, and G. Zhang. Fault-tolerant attitude determination and control system design of nanosatellite 2. *Journal of Aerospace Engineering*, 31(6), 2018.

[62] A. Si Mohammed. An attitude determination and control system of the alsat-1 first algerian microsatellite. 2003.

[63] A. Slavinskis, H. Ehrpais, H. Kuuste, I. Sünter, J. Viru, J. Kütt, E. Kulu, and M. Noorma. Flight results of estcube-1 attitude determination system. *Journal of Aerospace Engineering*, 29(1), 2016.

[64] J. Slotine. Slotine lectures on nonlinear systems. Nonlinear Systems Laboratory Massachusetts Institute of Technology, 2014.

[65] J. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Education Taiwan Ltd., 1991.

[66] J. Springmann and J. Cutler. Flight results of a low-cost attitude determination system. *Acta Astronautica*, 99(1), 2014.

[67] J. Springmann, A. Sloboda, A. Klesh, M. Bennett, and J. Cutler. The attitude determination system of the rax satellite. *Acta Astronautica*, 75, 2012.

[68] W. Steyn and Y. Hashida. In-orbit attitude performance of the 3-axis stabilised snap-1 nanosatellite. *15th AIAA/USU Conference on Small Satellites, SSC01-V-1b*, 2004.

[69] W. Steyn, Y. Hashida, and V. Lappas. An attitude control system and commissioning results of the snap-1 nanosatellite. *Proceeding of 14th AIAA/USU Conference on Small Satellites*, 2000.

[70] M. Tafazoli. A study of on-orbit spacecraft failures. *Acta Astronautica*, 64(2-3), 2009.

[71] C. Van Beusekom, R. Lisowski, J. Fulton, and C. Morand. Three-axes attitude determination and control system design for low-cost micro-satellites. volume 6, 2003.

[72] S. Xie and X. Peng. Fault diagnosability of momentum wheel based on bayesian network. *Advanced Materials Research*, 694 697, 2013.

[73] G. Yanning, G. Youmin, G. Sheng, and M. Guangfu. The node selection of satellite attitude control system based on sensitivity function in bayesian network. 2016.

APPENDIX A

CONDITIONAL PROBABILITY DISTRIBUTIONS FOR THE BAYESIAN NETWORK

The conditional probability distributions have been presented based on variable numbers for the purpose of convenience in constructing the tables. These numbers have been illustrated in Figure 5.2.

1 = occurrence of a fault or error
0 = absence of a fault or error

Table A.1: Orbit Determination Fault, #1

1	
P(1=0)	0.95
P(1=1)	0.05

Orbit determination in the satellite is carried out using both Two-Line Element (TLE) and Global Navigation Satellite System (GNSS), resulting in a low probability of faults or errors occurring.

Table A.2: IGRF Model Fault, #2

2	
P(2=0)	0.95
P(2=1)	0.05

IGRF provides a mathematical model of the Earth's magnetic field, and the probability of faults or errors occurring in this model is low.

Table A.3: Incorrect Clock, #3

3	
P(3=0)	0.99
P(3=1)	0.01

Accurate timekeeping is typically maintained by computers; however, instances of incorrect clock readings can be observed when the electronic components are impacted by radiation.

Table A.4: Astronomical Almanac Model Fault, #4

4	
P(4=0)	0.99
P(4=1)	0.01

In the astronomical almanac model, the Sun is considered a point-like source, which is not entirely accurate. Additionally, due to the significant distance between the Earth and the Sun, the direction of the Sun's vector is determined by drawing a line from the Earth's center to the Sun. Consequently, the orbit of the satellite in this context is not of great significance. These factors can be regarded as potential shortcomings in the almanac model. Nevertheless, the likelihood of these shortcomings causing errors is minimal, as they can be considered in the level of sensor noise.

Table A.5: IGRF Fault,#5

5	P(5=0)	P(5=1)
1=0,2=0,3=0	0.95	0.05
1=0,2=0,3=1	0.9	0.1
1=0,2=1,3=0	0.1	0.9
1=0,2=1,3=1	0.1	0.9
1=1,2=0,3=0	0.05	0.95
1=1,2=0,3=1	0.05	0.95
1=1,2=1,3=0	0.02	0.98
1=1,2=1,3=1	0.01	0.99

Incorrect clock data has a relatively low impact on the IGRF. This is because the magnetic field's variation over time is relatively small when compared to the potential errors in orbit determination.

Table A.6: Astronomical Almanac Fault, #6

6	P(6=0)	P(6=1)
3=0,4=0	0.95	0.05
3=0,4=1	0.05	0.95
3=1,4=0	0.05	0.95
3=1,4=1	0.01	0.99

In the joint probability distribution for errors in the astronomical almanac, both model faults and incorrect clocks are equally important.

Table A.7: Magnetometer Reference Vector, #7

7	P(7=0)	P(7=1)
5=0	0.99	0.01
5=1	0.01	0.99

In this joint probability scenario, if there is a fault in the IGRF, there is a 99% likelihood of a fault or error occurring in the magnetometer reference vector.

Table A.8: Sun Sensor Reference Vector, #8

8	P(8=0)	P(8=1)
6=0	0.99	0.01
6=1	0.01	0.99

Within this joint probability scenario, the presence of a fault in the astronomical model is associated with a 99% probability of experiencing a fault or error in the sun sensor reference vector.

Table A.9: Calibration Sun Sensor, #9

9	
P(9=0)	0.9
P(9=1)	0.1

The calibration error in the sun sensor is influenced by the particular sensor being utilized. When the sensor quality is high, the probability of this error is low. However, for the sensor used in this thesis, a 10% sensor fault is assumed.

Table A.10: Albedo, #10

10	
P(10=0)	0.7
P(10=1)	0.3

Albedo effect probabilities are determined based on the specific orbit configuration. In the case of a sun-synchronous orbit, there is a 30% likelihood of experiencing an albedo fault. Generally, such faults are more likely to occur when the spacecraft is positioned near the Earth's poles.

Table A.11: Eclipse, #11

11	
P(11=0)	0.65
P(11=1)	0.35

Eclipses are determined based on the type of orbit. These probabilities are specifically provided for sun-synchronous orbits.

Table A.12: Sun Sensor Fault, #12

12	P(12=0)	P(12=1)
9=0,10=0,11=0	0.95	0.05
9=0,10=0,11=1	0.01	0.99
9=0,10=1,11=0	0.1	0.9
9=0,10=1,11=1	0.01	0.99
9=1,10=0,11=0	0.05	0.95
9=1,10=0,11=1	0.01	0.99
9=1,10=1,11=0	0.01	0.99
9=1,10=1,11=1	0.01	0.99

In this joint probability scenario, the occurrence of both an eclipse and an albedo event indicates that the sensor readings are likely to be faulty. However, calibration faults are relatively more forgiving in comparison.

Table A.13: Reaction Wheel Dipole, #13

13	
P(13=0)	0.05
P(13=1)	0.95

Reaction wheels typically operate continuously, and as a result, there is a 95% probability that they generate a dipole.

Table A.14: Measurement While Torque Rod Activating, #14

14	
P(14=0)	0.9
P(14=1)	0.1

In this table, a different logic applies. Typically, measurement is not taken when the torque rod is activated. However, there is still a 10% probability that measurement can be taken, primarily when the timing for both measurements and the torque rod activation is not set correctly.

Table A.15: Actuator Magnetic Inference, #15

15	P(15=0)	P(15=1)
14=0,13=0	0.99	0.01
14=0,13=1	0.05	0.95
14=1,13=0	0.01	0.99
14=1,13=1	0.01	0.99

The probability of magnetic interference is higher when the torque rod is active compared to when the reaction wheel generates a dipole.

Table A.16: Electric Circuits, #16

16	
P(16=0)	0.05
P(16=1)	0.95

All current-carrying devices produce magnetic interference.

Table A.17: Magnetic Inference, #17

17	P(17=0)	P(17=1)
16=0,15=0	0.95	0.05
16=0,15=1	0.01	0.99
16=1,15=0	0.01	0.99
16=1,15=1	0.001	0.999

In this joint probability scenario, magnetic interference can result from both electric circuits and actuator magnetic interference, and their relative probabilities are equal.

Table A.18: Calibration Magnetometer, #18

18	
P(18=0)	0.2
P(18=1)	0.8

The probability of magnetometer calibration fault or error is given as 80%. Even though calibration is performed on the ground, errors in calibration can still occur in orbit.

Table A.19: Magnetometer Fault, #19

19	P(19=0)	P(19=1)
18=0,17=0	0.95	0.05
18=0,17=1	0.01	0.99
18=1,17=0	0.05	0.95
18=1,17=1	0.01	0.99

The probability of a magnetometer fault is higher when magnetic interference occurs.

Table A.20: QUEST Fault, #20

20	P(20=0)	P(20=1)
7=0,8=0,12=0,19=0,21=0	0.99	0.01
7=0,8=0,12=0,19=0,21=1	0.001	0.999
7=0,8=0,12=0,19=1,21=0	0.05	0.95
7=0,8=0,12=0,19=1,21=1	0.001	0.999
7=0,8=0,12=1,19=0,21=0	0.001	0.999
7=0,8=0,12=1,19=0,21=1	0.001	0.999
7=0,8=0,12=1,19=1,21=0	0.001	0.999
7=0,8=0,12=1,19=1,21=1	0.001	0.999
7=0,8=1,12=0,19=0,21=0	0.001	0.999
7=0,8=1,12=0,19=0,21=1	0.001	0.999
7=0,8=1,12=0,19=1,21=0	0.001	0.999
7=0,8=1,12=0,19=1,21=1	0.001	0.999
7=0,8=1,12=1,19=0,21=0	0.001	0.999
7=0,8=1,12=1,19=0,21=1	0.001	0.999
7=0,8=1,12=1,19=1,21=0	0.001	0.999
7=0,8=1,12=1,19=1,21=1	0.001	0.999
7=1,8=0,12=0,19=0,21=0	0.05	0.95
7=1,8=0,12=0,19=0,21=1	0.001	0.999
7=1,8=0,12=0,19=1,21=0	0.05	0.95
7=1,8=0,12=0,19=1,21=1	0.001	0.999
7=1,8=0,12=1,19=0,21=0	0.001	0.999
7=1,8=0,12=1,19=0,21=1	0.001	0.999
7=1,8=0,12=1,19=1,21=0	0.001	0.999
7=1,8=0,12=1,19=1,21=1	0.001	0.999
7=1,8=1,12=0,19=0,21=0	0.001	0.999
7=1,8=1,12=0,19=0,21=1	0.001	0.999
7=1,8=1,12=0,19=1,21=0	0.001	0.999
7=1,8=1,12=0,19=1,21=1	0.001	0.999
7=1,8=1,12=1,19=0,21=0	0.001	0.999

Table A.20 – Continued from previous page

20	P(20=0)	P(20=1)
7=1,8=1,12=1,19=0,21=1	0.001	0.999
7=1,8=1,12=1,19=1,21=0	0.001	0.999
7=1,8=1,12=1,19=1,21=1	0.001	0.999

In the QUEST fault dominant parts are given with higher probability in the table.

Table A.21: Reference Vector Close to Each Other, #21

21	
P(21=0)	0.99
P(21=1)	0.01

Sometimes, reference vectors, such as the sun direction vector obtained from the almanac model and the magnetic field vector from the IGRF, can become closely aligned, with an angle between them of approximately 0 or 180 degrees. In such cases, these two vectors convey redundant information, leading to errors in the QUEST algorithm. This occurrence is influenced by the spacecraft's orbit and the season, as the sun pointing vector varies with changing seasons.

Table A.22: Inertia Uncertainty, #22

22	
P(22=0)	0.9
P(22=1)	0.1

The uncertainty in the inertia matrix is typically assumed to be around 10%.

Table A.23: Other Disturbances, #23

23	
P(23=0)	0.9
P(23=1)	0.1

In space, various disturbance torques can affect spacecraft. It is common practice to estimate disturbance torques as being approximately 10% of the magnitude of the acting torque.

Table A.24: Magnetic Residual Dipole, #24

24	
P(24=0)	0.1
P(24=1)	0.9

The occurrence of a magnetic residual dipole moment as a fault is expected with a probability of 90%.

Table A.25: Disturbance, #25

25	P(25=0)	P(25=1)
22=0,23=0,24=0	0.95	0.05
22=0,23=0,24=1	0.01	0.99
22=0,23=1,24=0	0.01	0.99
22=0,23=1,24=1	0.01	0.99
22=1,23=0,24=0	0.05	0.95
22=1,23=0,24=1	0.01	0.99
22=1,23=1,24=0	0.01	0.99
22=1,23=1,24=1	0.001	0.999

The effect of inertia uncertainty is taken into account during the design phase, and as a result, its impact on disturbances is relatively low.

Table A.26: Reaction Wheel Fault, #26

26	
P(26=0)	0.9
P(26=1)	0.1

A reaction wheel is a mechanically complex actuator, and there is a 10% probability of a fault occurring.

Table A.27: Torque Rod Fault, #27

27	
P(27=0)	0.98
P(27=1)	0.02

Torque rods are simpler actuators and, as a result, are less mechanically complex compared to reaction wheels. Therefore, the probability of a fault occurring in a torque rod is generally lower than that of a reaction wheel.

Table A.28: Actuator, #28

28	P(28=0)	P(28=1)
26=0,27=0	0.95	0.05
26=0,27=1	0.05	0.95
26=1,27=0	0.01	0.99
26=1,27=1	0.01	0.99

Actuator faults are most likely to occur when they are observed in the reaction wheel. This is because reaction wheels are typically used in main mission modes, making them more susceptible to faults.

Table A.29: Mathematical Model Mismatch, #29

29	P(29=0)	P(29=1)
25=0,28=0	0.95	0.05
25=0,28=1	0.05	0.95
25=1,28=0	0.05	0.95
25=1,28=1	0.01	0.99

Both disturbances and actuators have an equal effect on the mathematical model mismatch.

Table A.30: Insufficient Sampling Rate, #30

30	
P(30=0)	0.999
P(30=1)	0.001

The probability of insufficient sampling rate errors is very low.

Table A.31: Bias Instability Increase, #31

31	
P(31=0)	0.95
P(31=1)	0.05

The occurrence of bias instability increase can vary depending on the type of gyroscope used and the specific application it's employed in.

Table A.32: Calibration Gyro, #32

32	
P(32=0)	0.8
P(32=1)	0.2

Scale factor errors, misalignment errors, and orthogonality errors are typically considered as calibration errors in the gyroscope's measurements.

Table A.33: Variance Change, #33

33	
P(33=0)	0.95
P(33=1)	0.05

Changes in the angular random walk can be considered as variance changes. These changes can be influenced by factors such as temperature variations and the type of gyroscope being used.

Table A.34: Gyro Fault, #34

34	P(34=0)	P(34=1)
30=0, 31=0,32=0, 33=0	0.99	0.01
30=0, 31=0,32=0, 33=1	0.1	0.9
30=0, 31=0,32=1, 33=0	0.9	0.1
30=0, 31=0,32=1, 33=1	0.1	0.9
30=0, 31=1,32=0, 33=0	0.9	0.1
30=0, 31=1,32=0, 33=1	0.1	0.9
30=0, 31=1,32=1, 33=0	0.8	0.2
30=0, 31=1,32=1, 33=1	0.05	0.95
30=1, 31=0,32=0, 33=0	0.9	0.1
30=1, 31=0,32=0, 33=1	0.05	0.95
30=1, 31=0,32=1, 33=0	0.8	0.2
30=1, 31=0,32=1, 33=1	0.05	0.95
30=1, 31=1,32=0, 33=0	0.8	0.2
30=1, 31=1,32=0, 33=1	0.05	0.95
30=1, 31=1,32=1, 33=0	0.7	0.3
30=1, 31=1,32=1, 33=1	0.01	0.99

Estimation algorithms handle bias estimation, but there's no protection in place for variance changes. Consequently, when variance changes occur, the probability of gyro faults increases.

Table A.35: Lens Blocked due to Bright Object, #35

35	
P(35=0)	0.95
P(35=1)	0.05

During satellite detumbling or maneuvering, it is possible for the star tracker lens to become blocked by a bright object.

Table A.36: Calibration Star Tracker, #36

36	
P(36=0)	0.99
P(36=1)	0.01

Given that this is a small spacecraft, a star tracker calibration error of 1% is possible.

Table A.37: Image Contains Non-stellar Object, #37

37	
P(37=0)	0.95
P(37=1)	0.05

The star tracker has a 5% probability of detecting objects that are not stars.

Table A.38: Star Tracker Fault, #38

38	P(38=0)	P(38=1)
35=0,36=0,37=0	0.99	0.01
35=0,36=0,37=1	0.95	0.05
35=0,36=1,37=0	0.9	0.1
35=0,36=1,37=1	0.8	0.2
35=1,36=0,37=0	0.5	0.5
35=1,36=0,37=1	0.45	0.55
35=1,36=1,37=0	0.4	0.6
35=1,36=1,37=1	0.1	0.9

In this table, when the lens is blocked due to a bright object, it's possible that not all of the lenses are covered by the bright object. Typically, star tracker manufacturers state that if half of the lenses are blocked with a probability of 50%, the true value of the attitude can be determined. This particular scenario is represented in the table as 35=1,36=0,37=0 row. The remaining joint probabilities in the table have been determined based on careful consideration of the star tracker's performance.

Table A.39: Graph#1

Graph#1 = Quest and Gyro		
39	P(39=0)	P(39=1)
20=0,34=0	0.95	0.05
20=0,34=1	0.9	0.1
20=1,34=0	0.1	0.9
20=1,34=1	0.01	0.99

Table A.40: Graph#2

Graph#2 = Star Tracker and Gyro		
39	P(39=0)	P(39=1)
38=0,34=0	0.99	0.01
38=0,34=1	0.9	0.1
38=1,34=0	0.1	0.9
38=1,34=1	0.01	0.99

Table A.41: Graph#3

Graph#3 = Quest and Mathematical Model		
39	P(39=0)	P(39=1)
20=0,29=0	0.95	0.05
20=0,29=1	0.7	0.3
20=1,29=0	0.1	0.9
20=1,29=1	0.01	0.99

Table A.42: Graph#4

Graph#4 = Star Tracker and Mathematical Model		
39	P(39=0)	P(39=1)
38=0,29=0	0.98	0.02
38=0,29=1	0.8	0.2
38=1,29=0	0.1	0.9
38=1,29=1	0.01	0.99