The Advantages of Left-Invariant Extended Kalman Filtering for Spacecraft Attitude Estimation

Parth Shrotri

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2026

Committee:

Mehran Mesbahi

Amir Taghvaei

Program Authorized to Offer Degree:
Aeronautics and Astronautics

University of Washington

Abstract

The Advantages of Left-Invariant Extended Kalman Filtering for Spacecraft Attitude Estimation

Parth Shrotri

Chair of the Supervisory Committee:
Mehran Mesbahi
William E. Boeing Department of Aeronautics and Astronautics

Accurate attitude estimation is a critical capability for small satellite missions, where size, weight, and power constraints limit sensor quality and computational resources. The Multiplicative Extended Kalman Filter (MEKF) is the industry-standard algorithm for attitude estimation but lacks theoretical guarantees of convergence, making it vulnerable to divergence under large initialization errors and infrequent measurements. The Invariant Extended Kalman Filter (IEKF) is a variant of the Extended Kalman Filter for systems whose state space is a Lie group that has local stability guarantees unavailable to the MEKF. This thesis demonstrates that the spacecraft attitude estimation problem satisfies the conditions required to apply the IEKF framework and derives the resulting Left-Invariant Extended Kalman Filter (LIEKF) using a rate-gyroscope and star tracker operating at different sampling rates. The MEKF and LIEKF are evaluated in simulation across stable and unstable dynamic conditions under varying initial attitude errors, gyroscope bias drift rates, and star tracker measurement rates. Under nominal conditions both filters perform comparably, but under adverse conditions the MEKF is susceptible to divergence while the LIEKF maintains a reliable state estimate. Monte Carlo analysis demonstrates that the LIEKF consistently converges more reliably than the MEKF as measurement availability decreases. As the LIEKF imposes no additional computational cost over the MEKF, it is a compelling drop-in replacement for small satellite attitude estimation applications.

TABLE OF CONTENTS

Page

LIST OF FIGURES

GLOSSARY

ADCS: Attitude Determination and Control System, the spacecraft subsystem responsible for estimating and controlling the vehicle's orientation

DCM: Direction Cosine Matrix, a 3×3 rotation matrix that describes the orientation of one reference frame relative to another

QUATERNION: A four-parameter representation of attitude consisting of a scalar part and a three-dimensional vector part, commonly used in spacecraft attitude estimation

LIE GROUP: A mathematical group that is also a smooth manifold, whose group operations are compatible with its smooth structure. The set of unit quaternions forms a Lie group

LIE ALGEBRA: The tangent space to a Lie group at the identity element, used to locally linearize the group structure

EKF: Extended Kalman Filter, a recursive state estimator for nonlinear dynamic systems based on local linearization of the system dynamics and measurement models

MEKF: Multiplicative Extended Kalman Filter, a variant of the EKF for attitude estimation that parameterizes the attitude error as a multiplicative quaternion perturbation to preserve the unit-norm constraint

IEKF: Invariant Extended Kalman Filter, a variant of the EKF for systems whose state space is a Lie group that exploits the group structure to obtain local stability guarantees

LIEKF: Left-Invariant Extended Kalman Filter, a variant of the IEKF that uses left-invariant error definitions and is applicable when the system measurements are left-invariant

RMS: Root Mean Squared, a measure of the magnitude of a set of values computed as the square root of the mean of the squared values

INERTIAL FRAME: A non-rotating, non-accelerating reference frame in which Newton's laws of motion hold. In this work, the inertial frame is centered at the Earth's center of mass and is fixed with respect to the distant stars (see ICRF [1])

BODY FRAME: A reference frame fixed to and rotating with the spacecraft body, with its origin at the spacecraft center of mass

CAMERA FRAME: A reference frame fixed to the star tracker, with its origin at the camera's optical center and its z-axis aligned with the camera boresight vector

LINE-OF-SIGHT VECTOR: A unit vector expressing the direction from an observer to a target. In this work, line-of-sight vectors refer to the directions from the spacecraft to observed stars, expressed in either the inertial frame or the star tracker camera frame.

ATTITUDE DETERMINATION: The process of computing the orientation of a spacecraft from instantaneous sensor measurements using deterministic methods that require no a priori information about the system state

ATTITUDE ESTIMATION: The process of estimating the orientation of a spacecraft by recursively fusing sensor measurements over time using stochastic methods that incorporate the system dynamics and a priori state information

GNSS: Global Navigation Satellite System, a network of satellites that provides positioning and timing information to receivers on the ground or in space

ACKNOWLEDGMENTS

Over the course of my educational journey, I've learned the true meaning of the saying "it takes a village". Therefore I must begin this work by thanking my village, without without whom none of this would have been possible.

I would first like to express my gratitude to my advisor, Professor Mehran Mesbahi whose *guidance*, support, feedback, and teaching has been instrumental to my growth as an engineer. I've greatly enjoyed your lectures, sense of humor, and control-theoretic perspective on life philosophy. Thank you as well for encouraging me to explore this topic. I would also like to thank Professor Amir Taghvaei for serving on my thesis committee, providing feedback on this work, for his excellent teaching in the state estimation and nonlinear control courses, and for providing me with the resources to tackle this research topic.

To the Aeroscience and Flight Mechanics Division at NASA's Johnson Space Center, my time as an intern with you all contributed immeasurably to my growth as an engineer and helped me fulfill a lifelong dream in contributing to human spaceflight. Thank you for your friendship, support, and for being a home for space geeks like me. I eagerly look forward to the day that our work meets in lunar orbit, hopefully in the not too distant future. Special thanks go to my mentors, Dr. Davis Adams, Dr. Kevin Kobylka, and Dr. Paul McKee for taking me on as an intern, helping me *navigate* my early career and introduction to engineering research, and for making so much of my work possible. Your *guidance* has been invaluable to my professional and personal development.

To my RAIN Lab colleagues Joseph Pinkston, Ruiqi Li, Joalda Morancy, Justin Chang, Shiva Shakeri, Jackson Zhou, Yifu Wang, Rick Zhang, and Spencer Kraisler, I am grateful to you all for your friendship, technical discussions, feedback, lunches, and podcast sessions, which enriched my time in graduate school immeasurably. Thank you for welcoming me to the lab, creating a supportive environment, and putting up with my awful puns.

To my friends: Graduate school is as much an intellectual journey as a personal one, and I could not have navigated it without you. To those who celebrated my successes along the way, helped me through the difficult moments, shared excitement for space exploration, or helped me process life, the world we live in, or the challenges that face our industry, your friendship and support means more than I can express. To my Seattleite friends who joined me on the trails, for brunches, or over dinners, thank you for making this city home.

To my family, my grandparents, aunts, uncles, and cousins: your love and support mean more to me than I can ever adequately put into words. Thank you for nurturing my passion for aerospace, encouraging me to pursue a career in engineering, and for cheering me on throughout my journey. Above all, I am forever grateful for my parents, my आई-बाबा, Monica and Shailendra Shrotri, who provided me with every opportunity I could ever ask for and more. Thank you for pushing me when I needed it, helping me grow as an engineer and person, and supporting my dreams from day one.

We came all this way to explore the Moon, and the most important thing is that we discovered the Earth.

William Anders

Lunar Module Pilot, Apollo 8

Always do the things that might even scare you. The things that intimidate you. The things you think that maybe are beyond your reach. But actually, when you achieve them, you learn the most about yourself and you bring the most back to the world.

Christina Koch

Mission Specialist, Artemis II

Consider again that dot. That's here. That's home. That's us. On it everyone you love, everyone you know, everyone you ever heard of, every human being who ever was, lived out their lives. The aggregate of our joy and suffering, thousands of confident religions, ideologies, and economic doctrines, every hunter and forager, every hero and coward, every creator and destroyer of civilization, every king and peasant, every young couple in love, every mother and father, hopeful child, inventor and explorer, every teacher of morals, every corrupt politician, every "superstar", every "supreme leader", every saint and sinner in the history of our species lived there – on a mote of dust suspended in a sunbeam.

Carl Sagan

Excerpt from *Pale Blue Dot*

DEDICATION

To all of the engineers who came before me and inspired me to follow in their footsteps, and to the future engineers and explorers of the Artemis Generation who I hope to inspire.

Chapter 1
INTRODUCTION

"Why did we even believe you could navigate by the stars? You can't even read a compass."

Captain Sense of Direction

The small satellite industry has grown rapidly in the past decade, driven by reductions in launch costs and the availability of miniaturized spacecraft hardware. Small satellites now serve a wide variety of civil, commercial, and defense applications, including remote sensing, in-space servicing, as well as scientific and engineering research. Each of these applications places demands on the precision with which the spacecraft must orient itself: remote sensing payloads must be pointed at specific ground targets, in-space servicing satellites must track other spacecraft, and scientific instruments must be directed at objects of interest. Accurate pointing is also necessary for other mission critical tasks such as orbit maintenance, trajectory correction, power generation, and communication. A prerequisite to achieving precise pointing is reliable knowledge of the spacecraft's current orientation, also known as its attitude, as well as its angular velocity. The satellite subsystem responsible for meeting these requirements is the Attitude Determination and Control System (ADCS) [2]. A problem inherent to small satellite platforms is that their size, weight, power, and cost constraints result in significant limitations on the quality of the sensors and computing resources onboard, which complicates estimator design and performance.

Attitude determination or estimation are the processes by which knowledge of the spacecraft's orientation is produced using onboard sensor measurements. Attitude determination refers to the set of methods and algorithms which are deterministic, do not use a priori information about the state of the system, and compute attitude from instantaneous observations. Attitude *estimation* refers to the set of stochastic methods that incorporate the dynamics of the system and recursively fuse sensor measurements over time to produce a statistically optimal estimate of the orientation. These measurements are inherently affected by noise, bias, environmental perturbations, and other corrupting factors. When using these noise corrupted measurements in attitude determination algorithms, the resulting estimate often has high error and uncertainty motivating the use of attitude estimation algorithms [2].

The Kalman Filter [3] is an optimal estimator for linear dynamical systems. Given Gaussian process and measurement noise, the Kalman filter is guaranteed to produce the minimum mean square error estimate of the state. However, the dynamics of most real-world systems, including spacecraft attitude kinematics, are nonlinear and thus, the Kalman Filter cannot be used directly to estimate the state of these system. The Extended Kalman Filter (EKF) is an adaptation of the Kalman Filter which can estimate the state of nonlinear dynamic systems. The EKF uses linearizations of the dynamic and measurement model equations to estimate the evolution of the state of the system and its associated error covariance matrix over time [4]. Due to the nonlinear nature of the system, in general, the EKF lacks optimality or convergence guarantees. Despite this, due to its computational simplicity and suitability for real-time implementation, it is the de facto choice for aerospace applications [5]. The EKF assumes that the state space is a vector space, and therefore state estimate corrections are applied additively. This assumption does not hold for rotational states, which for spacecraft are most commonly represented using unit quaternions [6] which describe the surface of a 4-dimensional hyper-sphere. Therefore, applying the additive formulation of the EKF to estimate attitude will violate the unit-norm constraint and introduce geometric inconsistencies.

The Multiplicative Extended Kalman Filter (MEKF), as formulated by Lefferts, Markley, and Crassidis [7], addresses this by parameterizing the attitude error as a small multiplicative quaternion perturbation and applying EKF updates to estimate the error state rather than to the quaternion directly. The MEKF has become the standard approach to spacecraft attitude estimation and has extensive heritage in operational spacecraft. Despite its widespread use, the MEKF, similar to the EKF, lacks any convergence guarantees. In practice, the MEKF relies on low initial estimate error and frequent measurement updates to remain converged. When the initial state estimate error is large or the measurement updates are infrequent, the linearization on which the filter depends may be inaccurate, and the filter state estimate can diverge from the true state [8].

The sensor architecture used to estimate the attitude of the spacecraft typically pairs a rate-gyroscope with an absolute attitude sensor. The rate-gyroscope provides high-frequency, direct measurements of body-frame angular velocity, making it the natural sensor for both attitude control [2] as well as for continuous attitude propagation between discrete correction steps [4]. Rate-gyroscopes are also susceptible to time-varying measurement bias, which accumulates over time and degrades the accuracy of the propagated attitude estimate if left uncorrected [9]. By including an absolute sensor, we can derive corrections that prevent drift in the attitude and attitude rate estimate. Examples of absolute sensors include sun sensors, horizon sensors, magnetometers, Global Navigation Satellite System (GNSS) receivers, star trackers, as well as more general vision-based sensors such as cameras [10, 11, 12, 13].

Star trackers are perhaps the most well known and commonly used absolute sensors onboard modern spacecraft owing to their high precision compared to alternatives [14]. A star tracker operates by imaging the star field and detecting the pixel coordinates of star centroids within its field of view. In order to use these observations in the attitude estimation problem, each detected star must be matched to an entry in an onboard star catalog. This process is known as star identification and is achieved using algorithms such as those presented in these works [15, 16, 17, 18]. A further practical challenge in estimation arises from this process, as these algorithms are sensitive to measurement noise as well as false stars caused by debris or other spacecraft. This can cause misidentification or inconclusive identification, resulting in periods of time where there are no valid star tracker measurements available for attitude estimation [15]. For an estimator without convergence guarantees, measurement outages represent a significant vulnerability. Without frequent corrections, accumulated gyroscope bias error can cause the attitude estimate to diverge irrecoverably from the true state.

A recent development in the field has been the use of properties of group structure to better inform state estimation. The Invariant Extended Kalman Filter (IEKF), developed by Barrau and Bonnabel [19], is a specialized variant of the EKF and is for use with systems whose state space is a Lie group. Taking advantage of the structure of the state space through the use of invariant error definitions and the Lie exponential map, the IEKF has local stability guarantees that cannot be obtained for the EKF or MEKF [20]. Since the state space of the attitude of a spacecraft is a Lie group, it is possible to formulate an attitude estimator such that it inherits the local stability guarantees of the IEKF, yielding theoretically stronger convergence properties than the MEKF [21]. The IEKF framework has previously been applied to unmanned aerial vehicle navigation using GNSS/INS, as well as simultaneous localization and mapping applications using visual sensors for mobile robots [19]. Despite this, the application of the IEKF framework to the spacecraft attitude estimation problem is limited. Barrau and Bonnabel previously demonstrated the feasibility of this framework for the satellite estimation problem, but did not incorporate the effect of rate-gyroscope bias in their analysis [22, 21]. Gui and de Ruiter [23] expanded on this work by applying the IEKF framework to estimate the attitude and rate-gyroscope bias, under stable attitude dynamics , using a simulated satellite equipped with a rate-gyroscope paired with a magnetometer and a sun sensor. However, their analysis assumed a fixed measurement sampling rate and stable dynamics, leaving open the question of how the IEKF performs relative to the MEKF when measurement availability is reduced and dynamics are unstable.

1.1 Contributions

This thesis expands on the work presented by Gui and de Ruiter [23], presenting a simulation based comparison of the Multiplicative Extended Kalman Filter (MEKF) and the Left-Invariant Extended Kalman Filter (LIEKF) for spacecraft attitude estimation using a rate-gyroscope and star tracker operating at different rates. The primary contributions of this work are as follows:

1. **Demonstration of filter performance under unstable attitude dynamics.** Both filters are evaluated on scenarios involving rotation and tumbling about the intermediate principal axis. This regime of highly nonlinear and unstable dynamics represents a challenging evaluation case for an estimator. The results demonstrate that both filters are capable of tracking unstable state evolution when the star tracker sampling rate is sufficient but that the MEKF is susceptible to divergence when measurement availability is reduced. In the same scenario, the LIEKF is able to maintain a reliable state estimate.

2. **Monte Carlo analysis of filter convergence.** A set of Monte Carlo trials is conducted to quantify the probability of convergence for each filter under simultaneous adverse conditions: large initial attitude error, unstable rotational dynamics, elevated bias drift, and reduced star tracker measurement rates. These trials characterize the relative robustness of both filters and demonstrate that the LIEKF consistently converges more reliably than the MEKF as measurement availability decreases.

These contributions demonstrate that the local convergence properties of the LIEKF result in a more robust estimator than the traditionally used MEKF. As demonstrated, since the two filters differ only in their quaternion update step, these benefits come at no additional computational cost, making the LIEKF a compelling drop-in replacement for the MEKF in resource-constrained applications. This robustness is particularly relevant to the small satellite market, where improved convergence under sparse measurements may enable the use of lower-cost, lower-frequency sensors.

1.2 Thesis Structure

This thesis is organized into six chapters. Chapter 1 introduces the problem of spacecraft attitude estimation, the state-of-the-art algorithms and sensors used, previous relevant works, and outlines the primary contributions of this work.

Chapter 2 presents the relevant mathematical background necessary to understand the formulation of the attitude estimation problem, the methodologies, and algorithms analyzed.

Chapter 3 formally defines the necessary dynamics and measurement models needed to describe the system. It also discusses the role of the Kalman filter in attitude estimation, with a primary focus on the derivation and implementation of the two filters that are evaluated in this work.

Chapter 4 describes the simulation framework used to evaluate the performance of the proposed methods. This includes the dynamic scenarios considered, the sensor noise characteristics, and the filter tuning parameters.

Chapter 5 presents and analyzes the results obtained from the simulations, comparing the performance of the filters under various conditions.

Finally, Chapter 6 summarizes the key findings and impacts of this work and discusses potential directions for future research.

Chapter 2
THEORETICAL FOUNDATIONS

"How about sending me a fourth gimbal for Christmas."

Michael Collins, Apollo 11 Command Module Pilot

—upon being told their spacecraft is approaching gimbal lock

In this chapter, we will introduce the fundamental concepts necessary to understand the work presented in this thesis. We begin by discussing how quaternions are used to represent the attitude of a body and how to convert between quaternions and other common attitude representations. We also introduce several important quaternion operations and identities. We then provide an introduction to Lie groups and a demonstration that unit quaternions are a Lie group. Then in order to introduce the state estimation problem, we give a brief overview of the extended Kalman Filter (EKF) for a generalized dynamical system. We discuss its component matrices, the steps of the EKF algorithm, and its shortcomings. Finally, we introduce the Invariant Extended Kalman Filter, the conditions for its use, and its advantageous properties.

2.1 Attitude Representations and Quaternions

A common way to represent the attitude of a spacecraft is to use the unit quaternion which is a singularity-free parameterization of attitude. In the aerospace industry, it is common to use the scalar last convention, as shown below:

$$\boldsymbol{q} = \begin{bmatrix} q_x & q_y & q_z & q_s \end{bmatrix}^\top = \begin{bmatrix} q_{\text{vec}} & q_s \end{bmatrix}^\top \tag{2.1}$$

The quaternion can also be derived from the axis-angle representation of the attitude:

$$\boldsymbol{q} = \begin{bmatrix} v\sin(\alpha/2) & \cos(\alpha/2) \end{bmatrix}^\top \tag{2.2}$$

Where α is the angle to rotate the coordinate system by and v is the vector about which to rotate the coordinate system. Another common attitude representation is the direction-cosine matrix (DCM) which can be constructed from a quaternion using the following formula:

$$\boldsymbol{A(q)} = \begin{bmatrix} \left(q_s^2 + q_x^2 - q_y^2 - q_z^2\right) & 2\left(q_x q_y + q_s q_z\right) & 2\left(q_x q_z - q_s q_y\right) \\ 2\left(q_x q_y - q_s q_z\right) & \left(q_s^2 - q_x^2 + q_y^2 - q_z^2\right) & 2\left(q_y q_z + q_s q_x\right) \\ 2\left(q_x q_z + q_s q_y\right) & 2\left(q_y q_z - q_s q_x\right) & \left(q_s^2 - q_x^2 - q_y^2 + q_z^2\right) \end{bmatrix} \tag{2.3}$$

Successive rotations can be computed by two different quaternion multiplication operators, as defined below. The Hamiltonian quaternion multiplication operator often used in the mathematics community is represented with the following symbol: $\odot$. In aerospace applications, it is typical to use the Shuster convention operator [6], denoted as: $\otimes$.

$$\boldsymbol{p} \odot \boldsymbol{q} = \boldsymbol{q} \otimes \boldsymbol{p} = \begin{bmatrix} q_s p_{\text{vec}} + p_s q_{\text{vec}} + p_{\text{vec}} \times q_{\text{vec}} \\ p_s q_s - {p_{\text{vec}}}^\top q_{\text{vec}} \end{bmatrix} \tag{2.4}$$

A pure quaternion is a quaternion with zero scalar part. When given a pure quaternion $\boldsymbol{a}$ and a unit quaternion $\boldsymbol{q}$ we can write the following:

$$\boldsymbol{a} \otimes \boldsymbol{q} = \Omega(a)\boldsymbol{q} \tag{2.5}$$

$$\boldsymbol{q} \otimes \boldsymbol{a} = \Gamma(a)\boldsymbol{q} \tag{2.6}$$

where

$$\Omega(\boldsymbol{a}) = \begin{bmatrix} -[a\times] & a \\ -a^\top & 0 \end{bmatrix} \quad \text{and} \quad \Gamma(\boldsymbol{a}) = \begin{bmatrix} [a\times] & a \\ -a^\top & 0 \end{bmatrix} \tag{2.7}$$

and $a\times$ represents the skew-symmetric matrix formed from the vector part of the quaternion $\boldsymbol{a}$. The inverse of a unit quaternion $\boldsymbol{q}$ is given by:

$$\boldsymbol{q}^{-1} = \begin{bmatrix} -q_{\text{vec}} & q_s \end{bmatrix}^\top \tag{2.8}$$

We can also define the difference between two quaternions $\boldsymbol{q}$ and $\hat{\boldsymbol{q}}$ using the multiplicative error:

$$\boldsymbol{\delta q} = \boldsymbol{q} \otimes \hat{\boldsymbol{q}}^{-1} \tag{2.9}$$

When $\boldsymbol{q} = \hat{\boldsymbol{q}}$:

$$\boldsymbol{\delta q} = \boldsymbol{q} \otimes \hat{\boldsymbol{q}}^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^\top \coloneqq \boldsymbol{q}_I$$

The time derivative of the inverse of a quaternion can be found using this property. Starting from the fact that:

$$\boldsymbol{q} \otimes \boldsymbol{q}^{-1} = \boldsymbol{q}_I$$

Taking the time derivative, we have:

$$\frac{d}{dt}\left(\boldsymbol{q} \otimes \boldsymbol{q}^{-1}\right) = \mathbf{0}$$

Applying the product rule, we have:

$$\boldsymbol{q} \otimes \frac{d}{dt}\left(\boldsymbol{q}^{-1}\right) + \frac{d}{dt}\left(\boldsymbol{q}\right) \otimes \boldsymbol{q}^{-1} = \mathbf{0}$$

Rearranging, we have:

$$\boldsymbol{q} \otimes \frac{d}{dt}\left(\boldsymbol{q}^{-1}\right) = -\frac{d}{dt}\left(\boldsymbol{q}\right) \otimes \boldsymbol{q}^{-1}$$

Right multiplying both sides of the equation by $\boldsymbol{q}^{-1}$, we have:

$$\boldsymbol{q}^{-1} \otimes \boldsymbol{q} \otimes \frac{d}{dt}\left(\boldsymbol{q}^{-1}\right) = -\boldsymbol{q}^{-1} \otimes \frac{d}{dt}\left(\boldsymbol{q}\right) \otimes \boldsymbol{q}^{-1}$$

The left-hand side now simplifies to:

$$\frac{d}{dt}\left(\boldsymbol{q}^{-1}\right) = -\boldsymbol{q}^{-1} \otimes \frac{d}{dt}\left(\boldsymbol{q}\right) \otimes \boldsymbol{q}^{-1} \tag{2.10}$$

2.2 Lie Groups

The following definitions are taken from Hall's book on *Lie Groups, Lie Algebras, and Representations* [24], which also presents proofs of the relevant theorems.

Matrix Lie Group

A matrix Lie group G is any subgroup of the general linear group $GL(n;\mathbb{C})$ with the following property: If A_m is a sequence of matrices in G and converges to a matrix A, then either A is in G or A is not invertible. All matrix Lie groups are Lie groups.

Matrix Lie Algebras

The matrix Lie algebra $\mathfrak{g}$ associated with a matrix Lie group G is defined as the set of all matrices X such that $\exp(tX)$ is in G for all real numbers t. Also, the matrix Lie algebra $\mathfrak{g}$ is the tangent space to the matrix Lie group G at the identity element, $\mathfrak{g}$ can be defined as the set of all derivatives of smooth curves in G that pass through the identity element. The Lie exponential map $\exp : \mathfrak{g} \to G$ maps elements from the Lie algebra to the Lie group. The logarithmic map $\log : G \to \mathfrak{g}$ maps elements from the Lie group to the Lie algebra.

Link between Quaternions, Lie Groups, and Lie Algebras

An important property of unit quaternions is that $\boldsymbol{q}$ and $-\boldsymbol{q}$ represent the same rotation in $SO(3)$. Thus, the set of unit quaternions is a "double cover" of $SO(3)$. Since $SO(3)$ is a Lie group [24], we can state that the set of unit quaternions forms a Lie group. An alternative way to see this is to construct the following complex matrix representation of a quaternion:

$$\mathbf{Q} = \begin{bmatrix} q_s + iq_x & q_y + iq_z \\ -q_y + iq_z & q_s - iq_x \end{bmatrix}$$

Taking the determinant of this matrix, we have the following:

$$\det(\mathbf{Q}) = q_s^2 + q_x^2 + q_y^2 + q_z^2 = \|\boldsymbol{q}\|^2$$

For unit quaternions, we have $\|\boldsymbol{q}\| = 1$. The set of all 2×2 complex matrices with unit determinant forms the special unitary group $SU(2)$ which is also a Lie group [24]. Thus, we can state that the set of unit quaternions is isomorphic to $SU(2)$ and thus forms a Lie group.

Since the Lie algebra is the tangent space to the Lie group at the identity element, we can compute the Lie algebra of unit quaternions by computing the tangent space at the identity quaternion $\boldsymbol{q}_I$. The tangent space at the identity element is given by:

$$T_{q_I} = \{v \in 4 : \boldsymbol{q}_I^\top v = 0\}$$

Thus, the Lie algebra of unit quaternions is given by:

$$\mathfrak{h} = \left\{ v \in \mathbb{R}^4 : v = \begin{bmatrix} v_x & v_y & v_z & 0 \end{bmatrix}^\top, v_x, v_y, v_z \in \mathbb{R} \right\}$$

This shows that the Lie algebra of unit quaternions is the set of pure quaternions.

The Lie Exponential Map of Quaternions

In order to map a pure quaternion $\boldsymbol{v}$ to a unit quaternion, we can use the following formula:

$$\exp_q(\boldsymbol{v}) = \begin{bmatrix} \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|} \sin(\|\boldsymbol{v}\|) & \cos(\|\boldsymbol{v}\|) \end{bmatrix} \tag{2.11}$$

2.3 The Extended Kalman Filter

The Extended Kalman Filter (EKF) is an observer for nonlinear dynamic systems. It uses a dynamic and measurement model of the system as well as sequential measurements of the system to produce a state estimate. We start with the following nonlinear system:

$$\dot{x}(t) = f(x(t), u(t), t) + G(t)w(t) \tag{2.12}$$

$$y(t) = h(x(t), t) + v(t) \tag{2.13}$$

where x is the state of the system, u is the control input to the system, y is the measurement of the system. $f(\cdot)$ is a nonlinear differentiable equation which is the time derivative of the state in the absence of noise. Likewise, $h(\cdot)$ nonlinear differentiable equation which maps states to measurements in the absence of noise. $G(\cdot)$ is a noise sensitivity function, and $w(t)$ and $v(t)$ are process and measurement noise, which are Gaussian with zero mean and covariance $Q(t)$ and $R(t)$ respectively. The Extended Kalman filter computes the estimate at time k, denoted $\hat{x}_k$ and the estimate error covariance P_k as follows:

1. Initialize the state estimate and estimate error covariance

$$\hat{x}(t_0) = \hat{x}_0 \tag{2.14}$$

$$P_0 = \mathbb{E}\left[(x(t_0) - \hat{x}(t_0)) (x(t_0) - \hat{x}(t_0))^\top \right] \tag{2.15}$$

2. Propagate state and error covariance until next measurement arrives

$$\dot{\hat{x}}(t) = f(\hat{x}(t), u(t), t) \tag{2.16}$$

$$\dot{P}(t) = F(\hat{x}(t), t)P(t) + P(t)F^{\top}(\hat{x}(t), t) + G(t)Q(t)G^{\top}(t) \tag{2.17}$$

where $F\left(\hat{x}(t), t\right) = \frac{\partial f}{\partial x}\Big|_{\hat{x}(t)}$

3. Take the last propagated state and error covariance matrix as $\hat{x}_k^-$ and P_k^-, compute the gain matrix K_k, and use the measurement at time k to update the state estimate

$$K_k = P_k^- H_k^{\top}\left(\hat{x}_k^-\right)\left[H_k\left(\hat{x}_k^-\right)P_k^- H_k^{\top}\left(\hat{x}_k^-\right) + R_k\right]^{-1} \tag{2.18}$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k\left[y_k - h\left(\hat{x}_k^-\right)\right] \tag{2.19}$$

$$P_k^+ = \left[I - K_k H_k\left(\hat{x}_k^-\right)\right]P_k^- \tag{2.20}$$

where $H_k\left(\hat{x}_k^-\right) = \frac{\partial h}{\partial x}\Big|_{\hat{x}_k^-}$

In the propagation step, the EKF uses the nonlinear system dynamics to predict the next state. It then linearizes the dynamics of the system about the current state estimate to propagate the uncertainty. In the update step, the EKF uses the nonlinear measurement model to compute the predicted measurements. It then linearizes the measurement model about the current state estimate to compute the Kalman gain and update the state estimate and uncertainty. By using linearization, the EKF can handle a wide range of nonlinear systems. However, this approach means that there is no guarantee that the EKF estimated state will converge to the true state of the system. For example, if the system $f(x(t), u, t), h(x(t), t)$ is highly nonlinear and the EKF estimated state is far from the true state, the linearization $F(\hat{x}, t)$ and $H_k(\hat{x}_k^-)$ will introduce large error into the system causing the system to diverge. Thus, it is generally necessary for the initial state estimate error to be small and for measurements to be frequent in order to ensure that the cumulative effect of linearization is small.

2.4 The Invariant Extended Kalman Filter

This section introduces the Invariant Extended Kalman Filter (IEKF) as described by Bonnabel and Barrau [19]. The IEKF is a specialized variant of the EKF which operates on a state space that is a Lie group. Using properties of Lie groups, the IEKF theoretically has local stability properties in terms of the error of the system. That is, regardless of the true state of the system, the estimate error will remain bounded. We can use two error definitions: the left- and right-invariant errors. If we define a state χ which lies in a Lie group and an estimated state $\hat{\chi}$, then the left- and right invariant errors are given by Equations (2.21) and (2.22), respectively.

$$\eta^L(t) = \chi(t)^{-1}\hat{\chi}(t) \tag{2.21}$$

$$\eta^R(t) = \hat{\chi}(t)\chi(t)^{-1} \tag{2.22}$$

In order to use the IEKF, we require that the state space of the system be a Lie group, that either the left- or right-invariant error dynamics can be written in a state trajectory independent form, and that the measurements are either left or right invariant. The left- and right-invariant errors have state trajectory independent propagation if the error dynamics can be written in the following form:

$$\frac{d}{dt}\eta(t) = g_u(\eta(t)) \tag{2.23}$$

where u is an exogenous input to the system. Either set of invariant error dynamics having state trajectory independent propagation is sufficient to satisfy this condition. The noiseless measurements are left- or right-invariant if they are of the form shown in Equations (2.24) and (2.25) respectively

$$Y = \chi d \tag{2.24}$$

$$Y = \chi^{-1} d \tag{2.25}$$

where d is a known vector. If all conditions are met, the IEKF can be formulated with the following propagation and state update steps:

1. Propagation

$$\dot{\hat{\chi}}(t) = g_{u(t)}(\hat{\chi}(t), t) \tag{2.26}$$

2. Left Invariant Update (LIEKF)

$$\hat{\chi}_k^+ = \hat{\chi}_k^- \exp\left(K_k \begin{bmatrix} \hat{\chi}^{-1} Y_1 - d_1 \\ \vdots \\ \hat{\chi}^{-1} Y_n - d_n \end{bmatrix}\right) \tag{2.27}$$

or Right Invariant Update (RIEKF)

$$\hat{\chi}_k^+ = \exp\left(K_k \begin{bmatrix} \hat{\chi} Y_1 - d_1 \\ \vdots \\ \hat{\chi} Y_n - d_n \end{bmatrix}\right) \hat{\chi}_k^- \tag{2.28}$$

where the Kalman gain K is computed using the Ricatti equation as before and $\exp(\cdot)$ is the Lie exponential map. Under the conditions discussed, the Lie logarithm of the error obeys a linear differential equation which yields the local stability properties of the IEKF [20].

Chapter 3
PROBLEM FORMULATION

"Essentially, all models are wrong, but some are useful."

George E.P. Box, Statistician

In this chapter, we introduce the attitude estimation problem and how it is formulated. We begin by introducing the mathematical models that describe the motion of the spacecraft as well as the measurement models for the onboard sensors. Next, we introduce static attitude determination methods as a means of estimating the attitude of a spacecraft using measurements taken at one instant in time. In order to estimate the attitude of the spacecraft over time using a sequence of measurements, we formulate the attitude estimation problem as an attitude error estimation and correction scheme. This forms the basis for the Multiplicative Extended Kalman Filter (MEKF) and the Left-Invariant Extended Kalman Filter (LIEKF). Finally, we discuss the details of the implementation and algorithm structure.

3.1 System Description and Models

In this work, the spacecraft is modeled as a single rigid body undergoing rotational dynamics. The attitude of the spacecraft describes the rotation from the inertial frame to the spacecraft body frame. In order to sense the rotational motion of the spacecraft, there are two types of sensors mounted to the spacecraft body: a rate-gyroscope, which provides body-frame angular velocity measurements, and a star tracker, which provides body-frame vector measurements of the direction to stars. In this section, we introduce the models necessary to formulate the spacecraft attitude estimation problem. Specifically, we require the spacecraft attitude kinematics and dynamics equations of motion that describe the evolution of the attitude over time, as well as the measurement models for the rate-gyroscope and the star tracker.

3.1.1 Attitude Kinematics and Dynamics

The attitude kinematics of the spacecraft, when represented by a unit quaternion, are given by Equation (3.1).

$$\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{\omega} \otimes \boldsymbol{q} \tag{3.1}$$

where $\boldsymbol{q}$ is the unit quaternion that represents the rotation from the inertial frame to the spacecraft's body frame. The pure quaternion $\boldsymbol{\omega}$ is formed from the angular velocity vector $\omega \in \mathbb{R}^3$ that represents the angular velocity of the body-frame with respect to the inertial frame, resolved in the body-frame and given in radians per second. The time derivative of $\dot{\omega}$ is given by Euler's rotational equation shown in Equation (3.2):

$$\dot{\omega} = J_b^{-1}\left(L_b - \omega \times (J_b\omega)\right) \tag{3.2}$$

where J_b is the 3×3 inertia tensor expressed in the body frame and $L_b \in \mathbb{R}^3$ is the external torque applied to the body and given in the body frame [2].

3.1.2 Star Tracker Measurement Model

A star tracker provides the location of star centroids in pixel coordinates in the camera frame. Given a star's line-of-sight vector in the inertial frame and the star tracker's attitude quaternion representing the rotation from the inertial frame to the camera frame, we can predict the location of a star in the camera frame. We can calculate the known line-of-sight vectors to stars using a catalog. The work presented in this thesis uses the Hipparcos catalog [25]. The line-of-sight vector to the j^{th} star in the inertial frame is calculated using the star's location on the celestial sphere, which is given in terms of its right ascension (α_j) and declination (δ_j) as shown in Figure 3.1. The star's cartesian coordinates in the inertial frame are given as:

$$s_j^I = \begin{bmatrix}\cos\delta_j \cos\alpha_j & \cos\delta_j \sin\alpha_j & \sin\delta_j\end{bmatrix}^\top \tag{3.3}$$

The line-of-sight vector to a star in the camera's body frame is given by:

$$s_j^B = \boldsymbol{A}(\boldsymbol{q}) s_j^I \tag{3.4}$$

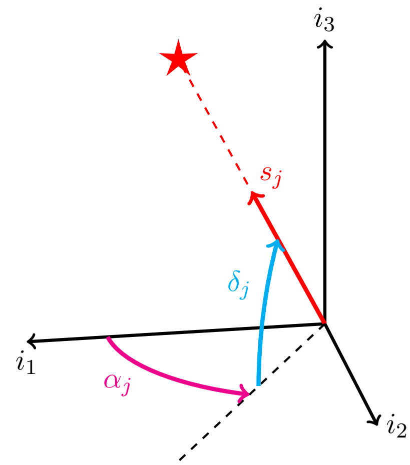


Figure 3.1: Line-of-Sight Vector to Star in Inertial Frame

where $\boldsymbol{q}$ is the attitude quaternion representing the rotation from the inertial frame to the camera's body frame. Introducing noise, the measurement of a star's line of sight vector is given by:

$$z_j^B = \boldsymbol{A}(\boldsymbol{q}) s_j^I + \nu_j \tag{3.5}$$

where ν_j is a zero-mean Gaussian process with covariance $\left(I_{3\times 3} - s_j^B s_j^{B^\top}\right)$ [4]. The star tracker outputs the centroid of a star in pixel coordinates. The relationship between the measured line-of-sight vector of an individual star and it's corresponding pixel coordinates is determined by using the linear mapping described by the pinhole camera model:

$$\begin{bmatrix} \mathrm{u}_j & \mathrm{v}_j & 1 \end{bmatrix}^\top = \frac{1}{K z_j^B[3]} K z^B \tag{3.6}$$

where u_j and v_j are the pixel coordinates of the j-th star and K is the camera intrinsic matrix [26]. This matrix or its inverse are found using a camera calibration algorithm [27], and can be applied directly to star pixel measurements to yield line-of-sight measurements. As mentioned in Chapter 1, before using these measurements in an estimator, the identity of the star must be determined using star identification algorithms. The implementation of the star tracker emulation used in this work assumes catalog matching can be accomplished accurately.

3.1.3 Gyroscope Measurement Model

A rate-gyroscope is a common sensor to measure the angular velocity of the spacecraft's body-frame. A rate-gyroscope's measurement includes a bias as well as noise. A common model which represents the relationship between the true angular velocity of the spacecraft and the angular velocity measured by its rate-gyroscopes as presented by Markley and Crassidis [2] is given by Equation (3.7)

$$\tilde{\omega} = \omega + \beta + \eta_\omega \tag{3.7}$$

where $\tilde{\omega}$ is the measured angular velocity, ω is the true angular velocity, and β is a bias vector. The noise term, η_ω, is drawn from a zero-mean Gaussian distribution with covariance $\sigma_w^2 I_{3\times3}$. The bias term is time-varying and often has complex noise characteristics [28]. A simplification that we can make is to define the bias-rate, $\dot{\beta}$, as a random variable η_b, which is a zero-mean Gaussian with covariance $\sigma_b^2 I_{3\times3}$:

$$\dot{\beta} = \eta_b. \tag{3.8}$$

The initial bias β_0 is drawn from a zero-mean Gaussian with covariance $\sigma_{\beta_0}^2 I_3$.

3.2 Static Attitude Determination

Static attitude determination is the process by which we can solve for the attitude of a spacecraft using vector measurements taken at a singular point in time without any a priori information about the system. We can perform this computation deterministically by solving the attitude determination problem posed by Wahba [29] in which a set of weighted vector measurements are used to solve for an attitude matrix $\boldsymbol{A}(\boldsymbol{q})$ that minimizes the following loss function:

$$L(\boldsymbol{A}(\boldsymbol{q})) = \frac{1}{2}\sum_{j=1}^{N} a_i \| z_j^B - \boldsymbol{A}(\boldsymbol{q}) s_j^I \|^2$$

where a is a weighting vector, z_j^B is the measured line-of-sight vector in the spacecraft body frame to a star, s_j^I is the known line-of-sight vector to the star. Davenport's q Method [30] is one such solution to Wahba's problem.

This method uses the equivalent loss function:

$$L(\boldsymbol{A(q)}) = \lambda_0 - \sum_{j=1}^{N} a_j \left(z_j^B\right)^\top \boldsymbol{A(q)} s_j^I$$

where λ_0 is the sum of all the weights a_j. Using a set of quaternion relations as shown in [2] we have the loss function:

$$L(q) = \lambda_0 - q^\top K(B) q$$

where $K(B)$ is a real symmetric matrix with zero trace:

$$K(B) = \begin{bmatrix} B + B^\top - \text{Tr(B)} I_3 & c \\ c^\top & \text{Tr(B)} \end{bmatrix}$$

where B is the attitude profile matrix:

$$B = \sum_{j=1}^{N} a_j z_j^B \left(s_j^I\right)^\top$$

and

$$c = \sum_{i=1}^{N} a_j \left(z_j^B \times s_j^I\right)$$

We can then take the eigendecomposition of $K(B)$, with the optimal quaternion estimate, $\hat{\boldsymbol{q}}$ given by the normalized eigenvector corresponding to the largest eigenvalue.

This attitude estimate, when corrupted with zero-mean noise, will have zero-mean error. We can also predict the uncertainty of this attitude estimate with an analytic covariance model using the QUEST measurement model [31]. The error covariance of the estimate calculated by Davenport's q method is given in the body frame:

$$P = (\lambda_0 I_3 - B_0)^{-1} M (\lambda_0 I_3 - B_0)^{-1} \tag{3.9}$$

where

$$M = \sum_{j=1}^{N} a_j^2 \sigma_j^2 \left[I_3 - s_j^B \left(s_j^B\right)^\top\right]$$

and

$$B_0 = \sum_{j=1}^{N} a_j s_j^B \left(s_j^B\right)^\top$$

Since this formulation requires knowledge of the true body-frame line-of-sight vectors s_j^B, we can instead use the prediction of these vectors by assuming that the estimated attitude is approximately the true attitude:

$$s_j^B \approx \boldsymbol{A}(\hat{\boldsymbol{q}}) s_j^I$$

Thus, we have the following equations for M and B_0:

$$M = \sum_{j=1}^{N} a_j^2 \sigma_j^2 \left[I_3 - \boldsymbol{A}(\hat{\boldsymbol{q}}) s_j^I \left(\boldsymbol{A}(\hat{\boldsymbol{q}}) s_j^I \right)^\top \right] \tag{3.10}$$

and

$$B_0 = \sum_{j=1}^{N} a_j \boldsymbol{A}(\hat{\boldsymbol{q}}) s_j^I \left(\boldsymbol{A}(\hat{\boldsymbol{q}}) s_j^I \right)^\top \tag{3.11}$$

3.3 Error State Kalman Filter Derivation

Although vector measurements from a star tracker are sufficient to estimate the spacecraft attitude at a singular instant in time, the true evolution of the attitude is a continuous time process. By incorporating attitude rate measurements from a rate-gyroscope, we can capture this continuous time evolution. As discussed in Section 3.1.3, the measurements supplied by rate-gyroscopes are subject to a time-varying bias that requires correction. Thus, in order to produce an estimate of the attitude of the spacecraft and the rate-gyroscope measurement bias using a priori information about the system and measurements from multiple sensors, we can formulate an Extended Kalman filter in the form presented in Section 2.3. The filter estimates the following state vector:

$$\chi = \begin{bmatrix} \boldsymbol{q} & \beta \end{bmatrix}^\top$$

where $\boldsymbol{q}$ is the attitude quaternion representing the rotation from the inertial frame to the spacecraft's body-frame and β is the rate-gyroscope measurement bias.

Instead of directly estimating the state of the system as one might with standard EKF, for attitude kinematics, it is easier to formulate an EKF in terms of the evolution of error over a short period of time. In this formulation, attitude is propagated using rate-gyroscope measurements and the error between the propagated attitude and the true attitude is directly estimated using the star-tracker measurements. This error-state estimate is subsequently used to correct the estimate for the full state of the system. In the case of attitude estimation, the error-state is defined as the multiplicative error between the true attitude quaternion $\boldsymbol{q}$ and the estimated attitude quaternion $\hat{\boldsymbol{q}}$ given by Equation (2.9). The rate-gyroscope bias is also estimated using a similar framework.

3.3.1 Applicability of IEKF Framework

In order to apply the IEKF methodology, we must demonstrate that the conditions for its use, described in Section 2.4 are met. First, we must prove that the state space is a Lie group. We previously proved in Section 2.2 that the set of unit quaternions forms a double cover of $SO(3)$. Therefore, the state space of the spacecraft attitude, $\mathbb{H}$, is a Lie group. The state space of the rate-gyroscope measurement bias is simply $\mathbb{R}^3$, which is also a Lie group. Looking at the state space of χ, we see that this is simply a direct product group, $\mathcal{G} = \mathbb{H} \times \mathbb{R}^3$. Since the direct product of two Lie groups is also a Lie group [24], we have satisfied the first requirement that the state space of the system must be a Lie group. Additionally, the group action for the complete state space is given by the following expression:

$$(\boldsymbol{q}_1, \beta_1) \circ (\boldsymbol{q}_2, \beta_2) = (\boldsymbol{q}_1 \otimes \boldsymbol{q}_2, \beta_1 + \beta_2)$$

We must also have that the dynamics of the system are group affine. That is, the system's error dynamics are independent of the true state and the estimated state trajectories. We begin by defining the right invariant error-state of the system:

$$\eta = \hat{\chi}\chi^{-1} = \begin{bmatrix} \boldsymbol{\delta q} & \delta\beta \end{bmatrix}^\top$$

where $\boldsymbol{\delta q}$ is the multiplicative error between the true attitude quaternion $\boldsymbol{q}$ and the estimated attitude quaternion $\hat{\boldsymbol{q}}$. The bias error, $\delta\beta$, is the difference between the true rate-gyroscope bias β and the estimated bias $\hat{\beta}$.

Taking the time derivative of the error state, we have the following:

$$\dot{\eta} = \begin{bmatrix} \dot{\hat{\boldsymbol{q}}} \otimes \hat{\boldsymbol{q}}^{-1} + \boldsymbol{q} \otimes \frac{d}{dt}\left(\hat{\boldsymbol{q}}^{-1}\right) & \dot{\beta} - \dot{\hat{\beta}} \end{bmatrix}^{\top}$$

Using the quaternion kinematics described by Equation (3.1), we have the following expression for the time derivative of the error quaternion:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\boldsymbol{\omega} \otimes \boldsymbol{q} \otimes \hat{\boldsymbol{q}}^{-1} + \boldsymbol{q} \otimes \frac{d}{dt}\left(\hat{\boldsymbol{q}}\right)^{-1}.$$

Using the definition for the time derivative of the inverse of a quaternion given by Equation (2.10), we have the following:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\boldsymbol{\omega} \otimes \boldsymbol{q} \otimes \hat{\boldsymbol{q}}^{-1} - \boldsymbol{q} \otimes \hat{\boldsymbol{q}}^{-1} \otimes \frac{d}{dt}\left(\hat{\boldsymbol{q}}\right) \otimes \hat{\boldsymbol{q}}^{-1}$$

Substituting the definition of the error quaternion, we have:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\boldsymbol{\omega} \otimes \boldsymbol{\delta q} - \boldsymbol{\delta q} \otimes \frac{d}{dt}\left(\hat{\boldsymbol{q}}\right) \otimes \hat{\boldsymbol{q}}^{-1}$$

Then substituting the kinematics of the attitude estimate given by:

$$\frac{d}{dt}\left(\hat{\boldsymbol{q}}\right) = \frac{1}{2}\hat{\boldsymbol{\omega}} \otimes \hat{\boldsymbol{q}}, \tag{3.12}$$

we have the following expression for the error quaternion:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\boldsymbol{\omega} \otimes \boldsymbol{\delta q} - \frac{1}{2}\boldsymbol{\delta q} \otimes \hat{\boldsymbol{\omega}} \otimes \hat{\boldsymbol{q}} \otimes \hat{\boldsymbol{q}}^{-1} = \frac{1}{2}\left(\boldsymbol{\omega} \otimes \boldsymbol{\delta q} - \boldsymbol{\delta q} \otimes \hat{\boldsymbol{\omega}}\right). \tag{3.13}$$

Now we can substitute the relationship between the true and measured angular velocity given by the rate-gyroscope measurement model given by Equation (3.7) and define the estimated angular velocity as: $\hat{\omega} = \tilde{\omega} - \hat{\beta}$. This results in the following:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\left(\left(\tilde{\boldsymbol{\omega}} - \boldsymbol{\beta} - \boldsymbol{\eta}_{\boldsymbol{\omega}}\right) \otimes \boldsymbol{\delta q} - \boldsymbol{\delta q} \otimes \left(\tilde{\boldsymbol{\omega}} - \hat{\boldsymbol{\beta}}\right)\right)$$

where $\tilde{\boldsymbol{\omega}}, \boldsymbol{\beta}, \hat{\boldsymbol{\beta}}$, and $\boldsymbol{\eta}_{\boldsymbol{\omega}}$ are pure quaternions formed from their respective vectorial quantities in $\mathbb{R}^3$. Distributing terms, we have:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\left(\tilde{\boldsymbol{\omega}} \otimes \boldsymbol{\delta q} - \boldsymbol{\delta q} \otimes \tilde{\boldsymbol{\omega}} - \boldsymbol{\beta} \otimes \boldsymbol{\delta q} + \boldsymbol{\delta q} \otimes \hat{\boldsymbol{\beta}} - \boldsymbol{\eta}_{\boldsymbol{\omega}} \otimes \boldsymbol{\delta q}\right)$$

Expanding the quaternion multiplication operations, we have:

$$\boldsymbol{\tilde{\omega}} \otimes \boldsymbol{\delta q} - \boldsymbol{\delta q} \otimes \boldsymbol{\tilde{\omega}} = \begin{bmatrix} \delta q_s \cdot \tilde{\omega} + \tilde{\omega} \times \delta q_{\text{vec}} \\ -\tilde{\omega} \cdot \delta q_{\text{vec}} \end{bmatrix} - \begin{bmatrix} \delta q_s \cdot \tilde{\omega} + \delta q_{\text{vec}} \times \tilde{\omega} \\ -\tilde{\omega} \cdot \delta q_{\text{vec}} \end{bmatrix} = \begin{bmatrix} 2\left(\tilde{\omega} \times \delta q_{\text{vec}}\right) \\ 0 \end{bmatrix}$$

$$\boldsymbol{\delta q} \otimes \boldsymbol{\hat{\beta}} - \boldsymbol{\beta} \otimes \boldsymbol{\delta q} = \begin{bmatrix} \delta q_s \hat{\beta} + \delta q_{\text{vec}} \times \hat{\beta} \\ -\delta q_{\text{vec}} \cdot \hat{\beta} \end{bmatrix} - \begin{bmatrix} \delta q_s \beta + \beta \times \delta q_{\text{vec}} \\ -\beta \cdot \delta q_{\text{vec}} \end{bmatrix} = \begin{bmatrix} -\delta q_s \ \delta\beta - \left(\beta + \hat{\beta}\right) \times \delta q_{\text{vec}} \\ \delta q_{\text{vec}} \cdot \delta\beta \end{bmatrix}$$

Thus, the time derivative of the error quaternion can be written as follows:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\left(\begin{bmatrix} 2\tilde{\omega} \times \delta q_{\text{vec}} - \delta q_s \ \delta\beta - \left(\beta + \hat{\beta}\right) \times \delta q_{\text{vec}} \\ \delta q_{\text{vec}} \cdot \delta\beta \end{bmatrix} - \boldsymbol{\eta_{\omega}} \otimes \boldsymbol{\delta q}\right)$$

This can be simplified to the following:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\left(\begin{bmatrix} \left(2\tilde{\omega} - \beta - \hat{\beta}\right) \times \delta q_{\text{vec}} - \delta q_s \ \delta\beta \\ \delta q_{\text{vec}} \cdot \delta\beta \end{bmatrix} - \boldsymbol{\eta_{\omega}} \otimes \boldsymbol{\delta q}\right)$$

Substituting the definitions of bias error and estimated angular velocity, we have:

$$\frac{d}{dt}\left(\boldsymbol{\delta q}\right) = \frac{1}{2}\left(\begin{bmatrix} \left(2\hat{\omega} - \delta\beta\right) \times \delta q_{\text{vec}} - \delta q_s \ \delta\beta \\ \delta q_{\text{vec}} \cdot \delta\beta \end{bmatrix} - \boldsymbol{\eta_{\omega}} \otimes \boldsymbol{\delta q}\right)$$

Since the true rate-gyroscope bias is modeled as a random walk and the filter assumes zero bias rate, the time rate of change of the bias error is simply:

$$\frac{d}{dt}(\delta\beta) = \eta_\beta$$

Therefore, we have the following equation for the propagation of the error state:

$$\dot{\eta} = \begin{bmatrix} \frac{1}{2}\left(\begin{bmatrix} \left(2\hat{\omega} - \delta\beta\right) \times \delta q_{\text{vec}} - \delta q_s \ \delta\beta \\ \delta q_{\text{vec}} \cdot \delta\beta \end{bmatrix} - \boldsymbol{\eta_{\omega}} \otimes \boldsymbol{\delta q}\right) \\ \eta_\beta \end{bmatrix}$$

which is of the form $\dot{\eta} = g_{\hat{\omega}}(\eta, \eta_\omega, \eta_\beta)$. However, $\hat{\omega}$ is not an exogenous input, since it is estimated using the bias estimate. Therefore, our dynamics are not strictly group affine. However, if we approximate a small bias error, then we have $\hat{\omega} \approx \omega$. Since the angular velocity of the system is exogenous, now we have error dynamics which are approximately group affine, a sufficent condition based on past work [19]. Therefore, we have met all the conditions to apply the IEKF to the estimation of this system.

3.3.2 Reformulating the Attitude Kinematics

We can now reformulate the attitude kinematics further into the error state form as presented by Junkins and Crassidis [4]. Starting from the definition of the time derivative for the multiplicative error between the true attitude quaternion $\boldsymbol{q}$ and the estimated attitude quaternion $\hat{\boldsymbol{q}}$ given by Equation (3.13) and substituting the definition of the angular velocity error $\delta\omega = \omega - \hat{\omega}$, we have the following:

$$\boldsymbol{\delta\dot{q}} = \frac{1}{2}\left(\hat{\boldsymbol{\omega}} \otimes \boldsymbol{\delta q} - \boldsymbol{\delta q} \otimes \hat{\boldsymbol{\omega}}\right) + \frac{1}{2}\boldsymbol{\delta\omega} \otimes \boldsymbol{\delta q}$$

Since $\hat{\boldsymbol{\omega}}$ is a pure quaternion, we can use Equation (2.5) and Equation (2.6) to write:

$$\boldsymbol{\delta\dot{q}} = \frac{1}{2}\left(\Omega(\hat{\boldsymbol{\omega}})\boldsymbol{\delta q} - \Gamma(\hat{\boldsymbol{\omega}})\boldsymbol{\delta q}\right) + \frac{1}{2}\boldsymbol{\delta\omega} \otimes \boldsymbol{\delta q}$$

Simplifying, we have $\Omega(\hat{\omega})\boldsymbol{\delta q} - \Gamma(\hat{\omega})\boldsymbol{\delta q} = \left(\Omega(\hat{\omega}) - \Gamma(\hat{\omega})\right)\boldsymbol{\delta q}$ which can be written out and simplified as:

$$\left(\Omega(\hat{\omega}) - \Gamma(\hat{\omega})\right)\boldsymbol{\delta q} = \begin{bmatrix} -2\left[\hat{\omega}\times\right] & 0 \\ -2\hat{\omega}^\top & 0 \end{bmatrix}\boldsymbol{\delta q} = -\begin{bmatrix} 2\left[\hat{\omega}\times\right]\boldsymbol{\delta q}_{\text{vec}} \\ 0 \end{bmatrix}$$

Thus, the derivative of the error quaternion can be written as:

$$\boldsymbol{\delta\dot{q}} = -\begin{bmatrix} \left[\hat{\omega}\times\right]\delta q_{\text{vec}} \\ 0 \end{bmatrix} + \frac{1}{2}\boldsymbol{\delta\omega} \otimes \boldsymbol{\delta q}$$

Using the axis angle definition of the quaternion from Equation (2.2), we can use the small angle approximation to linearize the system: $v\sin(\alpha/2) \approx v\alpha/2$ and $\cos(\alpha/2) \approx 1$. Now, defining $\delta\alpha = v\alpha/2$, we can rewrite the error quaternion dynamics in reduced order form:

$$\delta\dot{\alpha} = -\left[\hat{\omega}\times\right]\delta\alpha + \boldsymbol{\delta\omega} \otimes \boldsymbol{\delta q}$$

When we assume a small attitude error, we can approximate the last term as $\delta\omega$. Thus, we can now write the dynamics as follows.

$$\delta\dot{\alpha} = -\left[\hat{\omega}\times\right]\delta\alpha + \delta\omega$$

Now, working with the angular rate gyroscope, we take the measurement model from Equation (3.7). As stated in Section 3.3.1, the estimated angular velocity is given by:

$$\hat{\omega} = \tilde{\omega} - \hat{\beta}$$

and the estimated bias rate is assumed constant in the propagation step:

$$\dot{\hat{\beta}} = 0$$

This has the important consequence that the bias rate is not updated in the propagation step and can only be updated in the measurement correction step. We can now write the angular velocity error as:

$$\delta\omega = \omega - \hat{\omega} = \tilde{\omega} - \beta - \eta_\omega - \tilde{\omega} - \hat{\beta} = -(\delta\beta + \eta_\omega)$$

Thus, we can write the dynamic model for the error-state Kalman filter as follows:

$$\delta\dot{\alpha} = -\left[\hat{\omega}\times\right]\delta\alpha - (\delta\beta + \eta_\omega)$$

3.3.3 *Formulating the Propagation Step*

Writing this in the form of Equation (2.12), we have:

$$\dot{\tilde{x}} = F(\hat{x}, t)\tilde{x} + G(t)w(t)$$

where the error-state vector is: $\tilde{x} = \begin{bmatrix} \delta\alpha(t) & \delta\beta(t) \end{bmatrix}^\top$ and the noise vector is: $w(t) = \begin{bmatrix} \eta_\omega(t) & \eta_b(t) \end{bmatrix}^\top$. Thus, the relevant matrices necessary for propagation as seen in Equation (2.17) are given as follows:

$$F(\hat{\omega}, t) = \begin{bmatrix} -\left[\hat{\omega}(t)\times\right] & -I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}$$

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

$$Q(\Delta t) = \begin{bmatrix} \left(\sigma_w^2\Delta t + \frac{1}{3}\sigma_b^2\Delta t^3\right) I_{3\times3} & -\left(\frac{1}{2}\sigma_b^2\Delta t^2\right) I_{3\times3} \\ -\left(\frac{1}{2}\sigma_b^2\Delta t^2\right) I_{3\times3} & \left(\sigma_b^2\Delta t\right) I_{3\times3} \end{bmatrix}$$

3.3.4 Formulating the Measurement Correction Step

Starting from the camera measurement model described in Equation (3.5), we can concatenate n individual star line-of-sight measurements to get the following:

$$y_k = \begin{bmatrix} \boldsymbol{A}(\boldsymbol{q}) s_1^I \\ \vdots \\ \boldsymbol{A}(\boldsymbol{q}) s_n^I \end{bmatrix}\Bigg|_{t_k} + \begin{bmatrix} \nu_1 \\ \vdots \\ \nu_n \end{bmatrix}\Bigg|_{t_k}$$

This is of the form $y_k = h_k(x_k) + v_k$ which is the discrete time variant of Equation (2.13). The measurement noise covariance matrix is given by:

$$R = \text{diag}\begin{bmatrix} \sigma_1^2 I_{3\times3} & \dots & \sigma_n^2 I_{3\times3} \end{bmatrix}$$

Thus, we can write the measurement equation as follows:

$$h_k(\hat{\boldsymbol{q}}_k^-) = \begin{bmatrix} \boldsymbol{A}(\boldsymbol{q}) s_1^I \\ \vdots \\ \boldsymbol{A}(\boldsymbol{q}) s_n^I \end{bmatrix}\Bigg|_{t_k}$$

This measurement update step occurs only when new star tracker measurements are available. We take the most recent propagated attitude quaternion as $\hat{\boldsymbol{q}}^-$ and the most recent error state estimate as $\hat{\tilde{x}}^- = \begin{bmatrix} \delta\hat{\alpha}^- & \delta\hat{\beta}^- \end{bmatrix}^\top$. Then, the actual attitude of the spacecraft is given by $\boldsymbol{A}(\boldsymbol{q}) = \boldsymbol{A}(\boldsymbol{\delta q})\boldsymbol{A}(\hat{\boldsymbol{q}}^-)$. A first order approximation of the error attitude matrix is given by:

$$\boldsymbol{A}(\boldsymbol{\delta q}) \approx I_{3\times3} - [\delta\alpha\times]$$

Then the true and estimated star line-of-sight vectors are given by:

$$s_j^B = \boldsymbol{A}(\boldsymbol{q}) s_j^I \quad \text{and} \quad s_j^{B-} = \boldsymbol{A}(\hat{\boldsymbol{q}}^-) s_j^I$$

Then the pre-innovation measurement residual is given by:

$$s_j^B - s_j^{B-} = \boldsymbol{A}(\boldsymbol{q}) s_j^I - \boldsymbol{A}(\hat{\boldsymbol{q}}^-) s_j^I = \left[\boldsymbol{A}(\hat{q}^-) s_j^I \times\right] \delta\alpha$$

Thus, the measurement sensitivity matrix is given by:

$$H_k(\hat{\boldsymbol{q}}_k^-) = \left.\begin{bmatrix} [\boldsymbol{A}(\hat{\boldsymbol{q}}^-)s_1^I\times] & 0_{3\times 3} \\ \vdots & \vdots \\ [\boldsymbol{A}(\hat{\boldsymbol{q}}^-)s_n^I\times] & 0_{3\times 3} \end{bmatrix}\right|_{t_k}$$

Now we can compute the post-innovation error-state estimate:

$$\hat{\tilde{x}} = K_k\left[y_k - h_k(\hat{\boldsymbol{q}}_k^-)\right]$$

Then we can use the estimated error-state to correct the estimate of the state vector of the system. The rate-gyroscope bias estimate can be updated with the error state with the following:

$$\hat{\beta}^+ = \hat{\beta} + \delta\hat{\beta}$$

In order to update the attitude quaternion estimate using the estimated error vector $\delta\hat{\alpha}$, we can use two different update mechanisms. The Left-Invariant Quaternion Update utilizes the Lie exponential map and yields a filter in IEKF form. Alternatively, we can use its approximation: the Multiplicative Quaternion Update, which yields the MEKF [4].

Left-Invariant Quaternion Update

Noting that the star line of sight measurements are left-invariant and that the post-innovation error state estimate is of the same form as the input to the left-invariant update described in Section 2.4, we can use the left invariant update step to correct the attitude estimate. The LIEKF uses the following update law:

$$\hat{\boldsymbol{q}}_{\boldsymbol{k}}^+ = \exp_q(\delta\hat{\alpha}) \otimes \hat{\boldsymbol{q}}_k^- \tag{3.14}$$

where $\exp_q(\cdot)$ is the Lie exponential map for quaternions given by Equation (2.11). Although unit quaternions describe the surface of a 4-dimensional unit hyper-sphere, we can visualize this operation on a 3D sphere shown in Figure 3.2

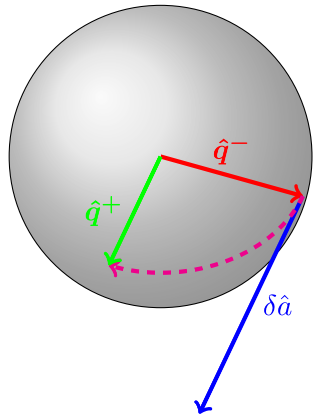


Figure 3.2: Visualization of Invariant Update

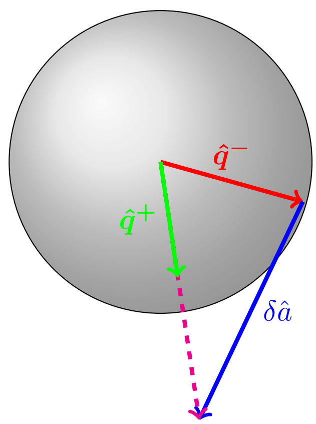


Figure 3.3: Visualization of Simplified Multiplicative Update

Multiplicative Quaternion Update

The attitude estimate can alternatively be computed using an approximation of the LIEKF update law. The MEKF uses the following update law to correct the attitude estimate:

$$\tilde{\boldsymbol{q}}_{\boldsymbol{k}}^{+} = \begin{bmatrix} \frac{1}{2}\delta\hat{\alpha}_k^+ \\ 1 \end{bmatrix} \otimes \hat{\boldsymbol{q}}_k^- \tag{3.15}$$

$\tilde{\boldsymbol{q}}_{\boldsymbol{k}}^{+}$ is then normalized to enforce the unit constraint which yields the new attitude estimate: $\hat{\boldsymbol{q}}_{\boldsymbol{k}}^{+}$. We can visualize this operation on a 3D sphere as shown in Figure 3.3.

3.4 Implementation of Filters for Attitude Estimation

The initial attitude estimate and error covariance for the filter can either be found automatically by applying Davenport's q method on the first available set of star line-of-sight measurements or by using an assumed attitude quaternion and initial error covariance. The initial bias estimate is assumed to be zero, and the initial error covariance of the bias estimate is chosen based on the standard deviation of the rate-gyroscope's initial bias.

Using the error-state Kalman filter formulation derived in Section 3.3 we can propagate the attitude estimate when rate-gyroscope measurements are available and when star line-of-sight vectors are available, we can estimate the error-state and correct the attitude quaternion estimate with either the invariant or multiplicative update. This algorithm is shown as a block diagram in Figure 3.4. When the invariant update is used, the algorithm described is the LIEKF and has local stability guarantees. When the simplified multiplicative update is used, the algorithm described is the MEKF.

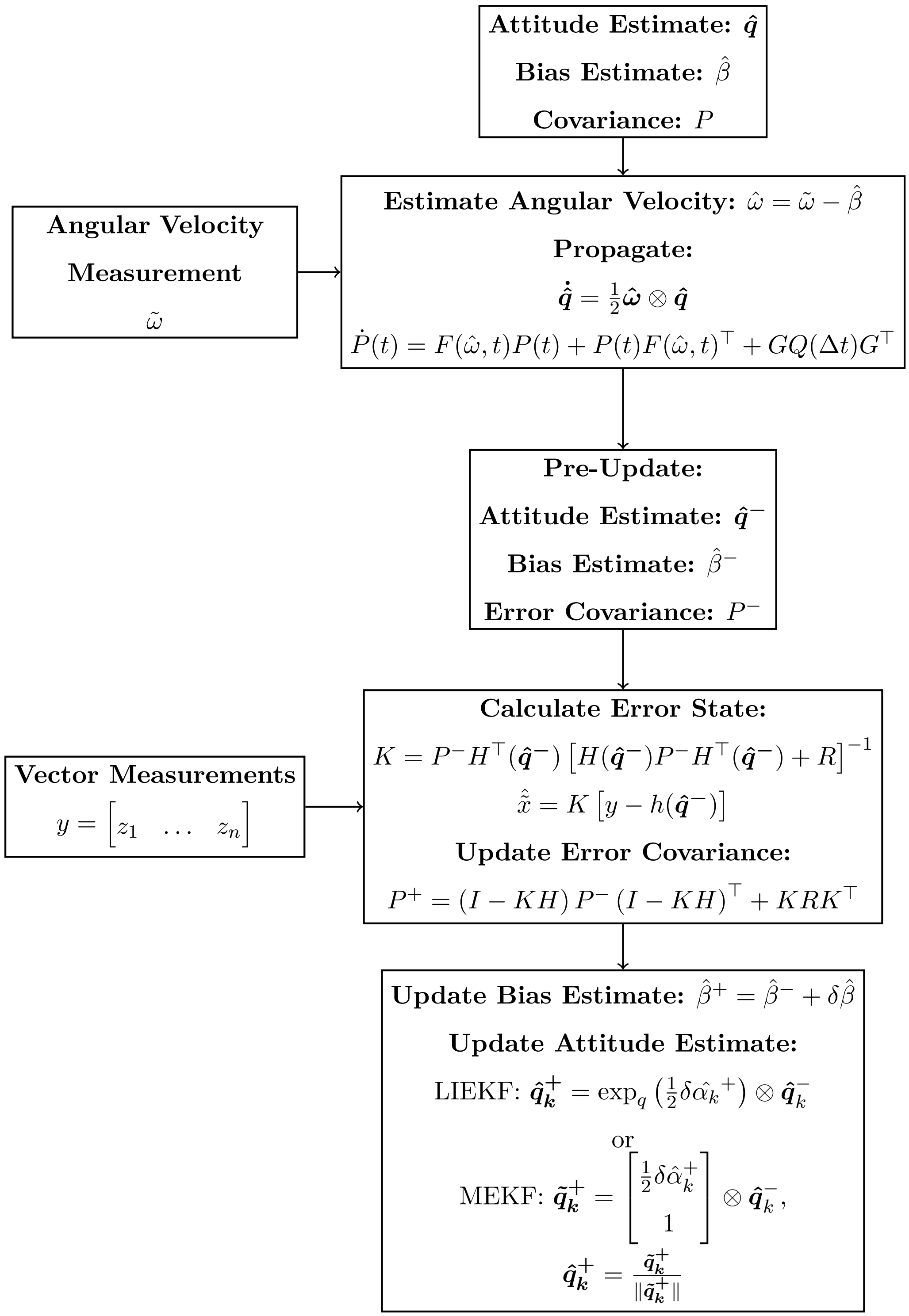


Figure 3.4: Structure of Error-State Kalman Filter Applied to Attitude Estimation

Chapter 4
SIMULATIONS

"When I wrote this code, only God and I knew how it worked. Now, only God knows it!"

Unknown Software Engineer

To evaluate the efficacy of both estimators in a variety of scenarios, a Python-based simulator was developed [32]. This simulator models the attitude dynamics of a spacecraft, its sensors, and an emulation of the onboard flight software. The attitude kinematics and dynamics of the spacecraft are described by the rigid-body equations of motion described in Section 3.1.1. Included in these dynamics is a control torque which is applied to the spacecraft such that it can be driven to a desired attitude and angular velocity.

The simulated star tracker and rate-gyroscope produce measurements according to the models described in Section 3.1.2 and Section 3.1.3, respectively. The emulated flight software contains the MEKF and LIEKF which uses measurements to update the attitude and rate-gyroscope bias estimates over time. The emulated flight software is also able to process user generated commands to change the attitude pointing and control modes to track different targets over the course of a simulation scenario. It also includes the proportional-derivative attitude controller, which determines the amount of torque to apply to the spacecraft to achieve the commanded state.

Each simulation trial is defined by a scenario configuration file and a directory of system configuration files. The scenario configuration file describes what the spacecraft will do over the course of the simulation. The system configuration files contain information about the spacecraft, its sensors, and the flight software parameters. The parameters contained in each set of files, as well as the simulator system architecture, the emulated flight software architecture, and a list of commands to modify the attitude control system mode and setpoints are provided in Appendix A.

4.1 Trial Scenarios

To evaluate the performance of the two attitude estimators under different dynamic conditions, four different simulation scenarios were devised. Each scenario is defined by its duration, time-step size, initial conditions, and the attitude control commands that the spacecraft will execute. An example of the parameters in a scenario configuration file are given in Table B.1.

4.1.1 Inertial-Hold

In the first scenario, the spacecraft maintains a fixed attitude with respect to the inertial frame. This scenario evaluates the convergence of the estimators in the presence of measurement noise and initial state estimate errors. The corresponding configuration parameters are listed in Table 4.1.

Duration	60 seconds
Initial Inertial-to-Body Quaternion	[0,0,0,1]
Initial Body Frame Angular Rate	(0.0,0.0,0.0) rad/sec
Commands	
Hold Attitude	• `time:` 0 seconds • `type: "GNC_ATT_CMD"` • `cmd: "Q(0,0,0,1)"`

Table 4.1: Inertial-Hold Scenario Parameters

4.1.2 Attitude Reconfiguration

In this dynamic scenario, the spacecraft first maintains its attitude for 60 seconds and then slews to a new orientation. This scenario evaluates the ability of the estimators to track attitude during a reconfiguration maneuver representative of typical spacecraft operations. The configuration parameters are given in Table 4.2.

Duration	200 seconds
Initial Inertial-to-Body Quaternion	[0,0,0,1]
Initial Body Frame Angular Rate	(0.0,0.0,0.0) rad/sec
Commands	
Hold Attitude	• `time: 0` seconds • `type: "GNC_ATT_CMD"` • `cmd: "Q(0,0,0,1)"`
Slew to New Attitude	• `time: 60` seconds • `type: "GNC_ATT_CMD"` • `cmd: "SPICE(3)"`

Table 4.2: Attitude Reconfiguration Scenario Parameters

4.1.3 Intermediate Axis Spin Up & Tumble

In this scenario, the spacecraft initially maintains its attitude for 60 seconds, after which it is commanded to rotate at a constant rate about its intermediate principal axis. The attitude controller is disabled, and a constant disturbance torque is applied for one second, causing the spacecraft to enter a passive tumble periodically changing its axis of rotation. This scenario evaluates estimator performance when tracking the unstable rotational dynamics associated with rotation about the intermediate principal axis. The configuration parameters are provided in Table 4.3.

4.1.4 Intermediate Axis Tumble

In this scenario, the spacecraft is initialized with a nonzero angular velocity about its intermediate principal axis. Initially, the spacecraft maintains its initial angular velocity for 10 seconds when it is forced into is passive tumble mode. This scenario evaluates estimator performance when initialized in an unstable dynamic regime. The configuration parameters are provided in Table 4.4.

4.2 System Configurations

The performance of nonlinear estimators are sensitive to initial state estimate errors, filter parameters, the sensor sampling rates, and other system specific configuration parameters. To determine the sensitivity of both attitude estimators, it is necessary to analyze their performance with different system configuration parameters. An example of the parameters in the system configuration are given in Tables B.2 to B.5.

4.2.1 Default

In this case, the measurement noise for both the rate-gyroscopes and star trackers are very low. The filters are initialized using the first set of star tracker measurements. Generally, navigation sensors on a spacecraft exhibit very low noise and have high sampling frequencies. Since there are several configuration parameters, only the relevant configuration parameters are given in Table 4.5.

4.2.2 Filter Initialization with Large Attitude Error

In this configuration, both filters are initialized with an attitude that is 180° from the true attitude. The error covariance of the initial state estimate $P(t = 0)$ is also increased to reflect the greater uncertainty in the initial state. The sensor parameters are unchanged from the default shown in Section 4.2.1. This configuration is designed to demonstrate the convergence properties of the filters when initialized in the worst case in terms of initial estimate error. Only the parameters modified from Table 4.5 are given in Table 4.6.

4.2.3 Filter Initialization with Large Attitude Error, Noisy & Sparse Measurements

This is a modification of the configuration described in Section 4.2.2, in which the standard deviation of the gyroscope bias random walk is increased and the star tracker sampling frequency is decreased. This configuration is designed to demonstrate the convergence properties of the filters when used with sub-optimal sensors. Only the parameters modified from Table 4.5 are given in Table 4.7.

4.3 Trial Success Criteria

In order to determine whether the MEKF or LIEKF has succeeded in estimating the state of the system for a specific scenario, we must define a set of criteria that accurately determines whether a filter has converged to the true state. The attitude estimation error, the difference between the true attitude and the estimated attitude, is given by Equation (2.9). The attitude error-state in both filters, $\delta\alpha$, has the following relationship to the attitude error quaternion:

$$\boldsymbol{\delta q} \approx \begin{bmatrix} \frac{1}{2}\delta\alpha & 1 \end{bmatrix}^\top$$

Thus, we can compute the attitude estimation error and approximate the error attitude vector as shown in Equation (4.1)

$$\delta\alpha \approx 2\boldsymbol{\delta q}_{\text{vec}} \tag{4.1}$$

Two sets of criterion are used to assess the convergence of a filter. The first method uses the root mean squared (RMS) of the attitude estimate-error vector. Since a large attitude estimate error is expected while the filter converges, particularly when the initial attitude estimate error is large, only the last 60 seconds of the simulation are considered when calculating the RMS of the attitude estimate error vector. If the RMS error for any axis exceeds a predefined threshold, the filter is considered to have failed. We set this threshold at 0.05 radians.

The second method assesses the consistency of the filter by requiring that the estimation error falls within the uncertainty bounds predicted by the filter. Specifically, the estimation error must fall within the $\pm 3\sigma$ bounds derived from the filter's estimate error covariance matrix P. For some sets of initial conditions, the estimation error is expected to temporarily exceed the $\pm 3\sigma$ bounds while the filter is converging. Therefore, time-based convergence criterion is necessary to classify the results of these trials. We use the following two criteria: The filter is considered to have failed if the attitude estimation error remains outside of its $\pm 3\sigma$ bounds for more than 20 seconds or if the rate-gyroscope bias estimation error remains outside of its $\pm 3\sigma$ bounds for more than 15 seconds.

Duration	120 seconds
Initial Inertial-to-Body Quaternion	[0,0,0,1]
Initial Body Frame Angular Rate	(0.0,0.0,0.0) rad/sec
Commands	
Hold Attitude	• `time:` 0 seconds • `type: "GNC_ATT_CMD"` • `cmd: "Q(0,0,0,1)"`
Spin-Up Around Intermediate Axis	• `time:` 60 seconds • `type: "GNC_ATT_CMD"` • `cmd: "RATE(0.0,1.0,0.0)"`
Turn Off Attitude Controller	• `time:` 70 seconds • `type: "GNC_ATT_CMD"` • `cmd: "OFF"`
Start Manual Torque Disturbance	• `time:` 80 seconds • `type: "GNC_ATT_CMD"` • `cmd: "TORQUE(100.0, 0.0, 0.0)"`
Stop Manual Torque Disturbance	• `time:` 81 seconds • `type: "GNC_ATT_CMD"` • `cmd: "AUTO_TORQUE"`

Table 4.3: Intermediate Axis Spin Up and Tumble Scenario Parameters

Duration	120 seconds
Initial Inertial-to-Body Quaternion	[0,0,0,1]
Initial Body Frame Angular Rate	(0.0,1.0,0.0) rad/sec
Commands	
Maintain Initial Angular Velocity	• `time:` 0 seconds • `type: "GNC_ATT_CMD"` • `cmd: "RATE(0.0,1.0,0.0)"`
Turn Off Attitude Controller	• `time:` 9 seconds • `type: "GNC_ATT_CMD"` • `cmd: "OFF"`
Start Manual Torque Disturbance	• `time:` 10 seconds • `type: "GNC_ATT_CMD"` • `cmd: "TORQUE(100.0, 0.0, 0.0)"`
Stop Manual Torque Disturbance	• `time:` 11 seconds • `type: "GNC_ATT_CMD"` • `cmd: "AUTO_TORQUE"`

Table 4.4: Intermediate Axis Tumble Scenario Parameters

Sensor Parameters Evaluated	
Gyro Bias Noise Standard Deviation	0.001 rad/sec
Gyro Bias Rate Standard Deviation	0.001 rad/sec^2
Gyro Sample Rate	50 Hz
Star Tracker Line-of-Sight Standard Deviation	1e-4 radians
Star Tracker Sample Rate	5 Hz
Filter Parameters Evaluated	
Initial State Estimate: $\hat{\boldsymbol{q}}(t=0), \hat{\beta}(t=0)$	• Attitude: `"AUTO"` • Bias: [0.0, 0.0, 0.0] rad/sec
Initial State Estimate Error Covariance: $P(t=0)$	• Attitude: `"AUTO"` • Bias: diag([0.1, 0.1, 0.1])
Process Noise Covariance Matrix: Q	diag([σ_w^2, σ_w^2, σ_w^2, σ_b^2, σ_b^2, σ_b^2]) • $\sigma_w = 0.01$ • $\sigma_b = 0.25$
Measurement Noise Covariance Matrix: R	$\sigma^2 I_{3n\times 3n}$ • $\sigma = 0.01$ • n = Number of star lines-of-sight measurements

Table 4.5: Default System Configuration

Filter Parameters Evaluated	
Initial State Estimate: $\hat{\boldsymbol{q}}(t=0)$, $\hat{\beta}(t=0)$	• Attitude: [1, 0, 0, 0] • Bias: [0.0, 0.0, 0.0] rad/sec
Initial State Estimate Error Covariance: $P(t=0)$	• Attitude: diag([9.86, 9.86, 9.86]) • Bias: diag([4.0, 4.0, 4.0])

Table 4.6: Filter Initialization with Large Attitude Error System Configuration

Sensor Parameters Evaluated	
Gyro Bias Rate Standard Deviation	0.01 rad/sec^2
Star Tracker Sample Rate	1, 0.5, or 0.33 Hz
Filter Parameters Evaluated	
Initial State Estimate: $\hat{\boldsymbol{q}}(t=0)$, $\hat{\beta}(t=0)$	• Attitude: [1, 0, 0, 0] • Bias: [0.0, 0.0, 0.0] rad/sec
Initial State Estimate Error Covariance: $P(t=0)$	• Attitude: diag([9.86, 9.86, 9.86]) • Bias: diag([4.0, 4.0, 4.0])

Table 4.7: Filter Initialization with Large Attitude Error, Noisy & Sparse Measurements System Configuration

Chapter 5
RESULTS AND DISCUSSION

"It doesn't matter how beautiful your theory is, it doesn't matter how smart you are. If it doesn't agree with experiment, it's wrong."

Richard Feynman, Theoretical Physicist

In this chapter, we present the results of the simulations performed and demonstrate the advantage that the LIEKF has over the MEKF. We begin by demonstrating how both filters perform when initialized with very low error in three different dynamic scenarios. The first two demonstrate the filters' performance under normal operating conditions. The last demonstrates how the filter performs when tracking unstable dynamics. Then, we demonstrate how both filters perform when initialized with a large attitude error. In these test cases, we verify that the filters can indeed correct large errors when the spacecraft dynamics are stable. Then, we evaluate their performance when the filters are initialized with large error and the spacecraft is initialized in an unstable dynamic regime. We follow this with an example demonstrating the susceptibility of the MEKF to diverge. Finally, we present a Monte Carlo analysis that quantifies the probability of convergence for each filter across a range of star tracker measurement rates, providing a characterization of the robustness advantage of the LIEKF over the MEKF.

To analyze how both filters performed, we analyze the attitude estimate error versus time for both the MEKF and the LIEKF as well as their corresponding $\pm 3\sigma$ bounds generated by their respective filter's error covariance estimate. The same is performed with the rate-gyroscope bias estimate.

5.1 Simulations with Default Parameters

This set of simulations uses the system configuration given in Section 4.2.1. The MEKF and LIEKF attitude estimates are initialized using the first available set of star tracker measurements. The rate-gyroscope bias is assumed to be zero at initialization, the measurement noise in both the rate-gyroscope and star trackers is fairly low, and the star tracker measurements are sampled at 5 Hz.

5.1.1 Inertial-Hold

This simulation uses the trial configuration given in Section 4.1.1. Since the attitude is unchanging and the attitude estimates were initialized with very little error, we do not expect to see any notable variation in the attitude error time history. Since it is possible that the rate-gyroscope bias was initialized with some error, we expect to see convergence to zero error over the course of the simulation. Also, since we expect the attitude error to be tightly bounded, we do not expect to see any notable variation between the attitude estimates provided by both filters.

The plots of the attitude estimation error versus time are given in Figure 5.1. We see that the attitude estimates provided by both filters never exceed their corresponding $\pm 3\sigma$ bounds and that the state estimates provided by both filters are extremely similar, as expected. We also see that the estimation uncertainty is the greatest in the direction of the star tracker's boresight vector (star tracker Z-axis). The initial jump in attitude estimate error is due to the gyroscope bias estimate being incorrect for the first few sampled points in time. The RMS attitude estimate error for both filters, taken during the last 30 seconds of the simulation, is given in Table 5.1.

Analyzing the error plots of the gyroscope bias, given in Figure 5.2, we see that the initial estimate error is corrected very quickly by both filters and remains converged for the duration of the simulation. In both sets of plots, there is a visible sawtooth characteristic in the $\pm 3\sigma$ bounding traces, which is due to the fact that the rate-gyroscope and star tracker measurements are sampled at different rates.

Axis	MEKF	LIEKF
X	6.43e-5	6.43e-5
Y	6.23e-5	6.23e-5
Z	1.42e-4	1.42e-4

Table 5.1: RMS Attitude Estimation Error (radians) – Inertial-Hold

5.1.2 Attitude Reconfiguration

This simulation uses the trial configuration given in Section 4.1.2. The attitude is maintained for 60 seconds, after which the spacecraft begins a slow slew to a new attitude. For the first 60 seconds, this scenario is identical to the Inertial-Hold scenario, so we expect that the filter behavior in this phase of the simulation will be very similar to the results presented in Section 5.1.1. Once the slew begins, there is some change in the angular acceleration which will cause both filters to be unable to reliably differentiate between rate-gyroscope drift and the motion of the spacecraft. Thus, we expect to see some increase in the rate-gyroscope bias estimate error as well as an increase in uncertainty in the estimates.

The plots of the attitude estimation error versus time are given in Figure 5.3. In this trial, we see that the attitude estimates provided by both filters never exceed their corresponding $\pm 3\sigma$ bounds. We also see that there is no discernible difference between the attitude estimates between either filter. In the first 60 seconds, we see that the filter behaves similarly to the results shown in Section 5.1.1, as expected. Once the slew maneuver begins, we see that the attitude error remains at or near zero, although the error covariance bounds exhibit some oscillations due to degraded observability at some points during the slew. The RMS attitude estimate error for both filters, taken during the last 60 seconds of the simulation, is given in Table 5.2. Analyzing the estimation error of the gyroscope bias versus time for both filters, given in Figure 5.4, the error remains bounded and close to or at zero for the duration of the simulation with no degradation in gyroscope bias estimate observed.

Axis	MEKF	LIEKF
X	6.41e-05	6.41e-05
Y	6.07e-05	6.07e-05
Z	1.62e-04	1.62-04

Table 5.2: RMS Attitude Estimation Error (radians) – Slew

5.1.3 Intermediate Axis Spin Up & Tumble

This simulation uses the trial configuration given in Section 4.1.3. The attitude is maintained for 60 seconds, after which the spacecraft begins a slow slew to a new attitude. For the first 60 seconds, this scenario is identical to the Inertial-Hold case, so we expect that the filter behavior in this phase of the simulation will be similar to the results presented in Section 5.1.1. Using the results from the slew scenario, we can predict that as the spacecraft spins up to its commanded angular velocity, there may be some increase in the rate-gyroscope bias estimate error as well as some increase in uncertainty in the estimates. Once the spacecraft begins its passive tumble, we also expect that the gyroscope bias estimate will degrade, potentially causing degradation in the attitude estimate as well.

The plots of the attitude estimation error versus time are given in Figure 5.5. In the first 60 seconds, we see that the filter behaves similarly to the results shown in Section 5.1.1, as expected. Once the spacecraft begins to rotate, although the error often remains close to zero, there are often violations of the $\pm 3\sigma$ bounds. The time spans in which a state estimate error exceeded its $\pm 3\sigma$ bound are marked in red in the figure. It is important to note that the violations shown in both the X and Y axes are not continuous periods of time. These are discrete events, typically immediately after a star tracker measurement update. This behavior appears to be similar or identical between both filters. We can also note that there are more violations of the error bounds in the axis of rotation. The RMS attitude estimate error for both filters, taken during the last 60 seconds of the simulation, is given in Table 5.3.

Analyzing the error plots for the gyroscope bias estimate, given in Figure 5.6, we see that the error remains bounded. However, there is a significant estimation error ($\approx \pm 0.1$ rad/sec) present in both filters when the spacecraft is spinning and tumbling. This gyroscope bias estimation error is likely the source of the higher than expected attitude estimation estimation error. It is important to note that the gyroscope bias estimate can only be updated when star tracker measurements are available, and the attitude estimate is propagated using the estimated angular velocity. Due to the relatively infrequent update of the gyroscope bias estimate, error in this step results in error integration for the next several attitude estimates. Since this behavior is similar or identical between both filters, it is reasonable to assume that this is due to a fundamental limitation of the system in its current configuration.

Axis	MEKF	LIEKF
X	0.00178	0.00178
Y	0.00849	0.00849
Z	0.00306	0.00306

Table 5.3: RMS Attitude Estimation Error (radians) – Intermediate Axis Spin & Tumble

5.2 Simulations with Large Initial Attitude Estimate Error

This set of simulations uses the system configuration described in Section 4.2.2. The MEKF and LIEKF attitude estimates are initialized 180° from the true attitude, the worst case scenario for an attitude estimator. The rate-gyroscope bias is assumed to be zero at initialization, the sensor noise is fairly low, and the star tracker measurements are sampled at 5 Hz. These simulations demonstrate how the MEKF and LIEKF respond when there are large errors in the initial state estimate.

5.2.1 Inertial-Hold

This simulation uses the trial configuration described in Section 4.1.1. Since the attitude dynamics in this trial are stable, we anticipate that the state estimates will converge to the true value despite the fact that the initial state estimate error is large. Since both attitude and gyroscope bias estimate errors are present, we also expect that both filters will take more time to converge to the true state.

The plots of the attitude estimation error versus time are given in Figure 5.7. We can see that after an initial transient period, the attitude error converges to zero. During this convergence period, the attitude error is expected to violate the predicted error bounds. We also see that both the MEKF and LIEKF have the same convergence time (≈ 3 seconds). The RMS attitude estimate error for both filters, taken during the last 30 seconds of the simulation, is given in Table 5.4.

The error plots of the gyroscope bias estimate, given in Figure 5.8 also show similar behavior. After an initial transient convergence period of approximately 3-4 seconds, the error converges to zero and remains within the predicted bounds. Although the gyroscope bias estimate initially has very little error, we see a significant temporary increase in error (> 6.0 rad/sec) while the filters are converging. We also did not see any notable difference between the MEKF and LIEKF error in this trial.

Axis	MEKF	LIEKF
X	6.32e-05	6.32e-05
Y	6.056e-05	6.056e-05
Z	1.56e-04	1.56e-04

Table 5.4: RMS Attitude Estimate Error (radians) – Large Initial Attitude Estimate Error, Inertial-Hold

5.2.2 Intermediate Axis Tumble

This simulation uses the trial configuration described in Section 4.1.4. Here, the spacecraft is initialized with a non-zero angular velocity about its intermediate principal axis. Since the dynamics in this trial are unstable and the initial state estimates are far from the state, there is some possibility that the state estimates do not converge to the true state. However, due to the relatively high sampling rate of the star tracker measurements (5 Hz), we expect that the filters should converge to the true state.

The plots of the attitude estimation error versus time are given in Figure 5.9. We see that although the convergence period is longer than in the case discussed in Section 5.2.1, both the MEKF and LIEKF exhibit low estimation error, demonstrating that both filters can correct large estimate errors when driven by unstable attitude dynamics. Also, we see that the LIEKF takes longer to converge to the true attitude. Whereas the MEKF error converges within the error bound by 4.9 seconds, the LIEKF converges by 5.6 seconds. This is due to the fact that there are large corrections made in the star tracker measurement update step which causes some overshoot in the error, thought this may be as a result of suboptimal filter parameter tuning. After the filters have converged, error remains low and the results are similar to the results presented in Section 5.1.3. Although the figures seem to show regions where the attitude estimate error falls outside of the predicted error bounds, these are, in fact, individual points in time. In this scenario, both filters meet the success criteria defined in Section 4.3. The RMS attitude estimate error for both filters, taken during the last 60 seconds of the simulation, is given in Table 5.5.

The error plots of the gyroscope bias estimate, given in Figure 5.10, show a convergence behavior that is similar to the results presented in Section 5.2.1. Once the filters have converged, we see that the bias estimation error remains bounded. However, similar to the results presented in Section 5.1.3, there is a significant estimation error ($\approx \pm 0.1$ rad/sec) present in both filters when the spacecraft is spinning and tumbling.

Axis	MEKF	LIEKF
X	0.00182	0.00182
Y	0.00905	0.00905
Z	0.00313	0.00313

Table 5.5: RMS Attitude Estimation Error (radians) – Large Initial Attitude Estimate Error, Intermediate Axis Tumble

5.3 Demonstration of MEKF Divergence

In this section, an example is presented in which the MEKF significantly diverges from the true state, while the LIEKF maintains a reliable state estimate. Using information obtained from the trial discussed in Section 5.2.2, a trial was designed to demonstrate the conditions under which the MEKF diverges. The system configuration described in Section 4.2.3 is used, in which the initial attitude estimate for both filters is set 180° from the true attitude, the standard deviation of the bias random walk is increased relative to the default configuration, and the star tracker measurement rate is set to 0.5 Hz. The trial is performed using the scenario described in Section 4.1.4, in which the spacecraft is initialized with a spin about its intermediate principal axis and forced into an unstable dynamic mode. Together, these conditions create an environment that can cause a nonlinear estimator to diverge.

The plots of the attitude estimation error versus time are given in Figure 5.11. The MEKF exhibits clear divergence, with large, rapidly oscillating errors that frequently exceed the predicted covariance bounds. By contrast, the LIEKF has converged to the true attitude after approximately 20 seconds and remains converged for the remainder of the simulation. The RMS attitude estimate error for both filters, taken during the last 60 seconds of the simulation, is given in Table 5.6. From this we can clearly see that the MEKF attitude estimate is unreliable while the LIEKF demonstrates similar performance to previous trials.

We gain further insight into the results of this simulation by looking at the error plots of the gyroscope bias estimate, given in Figure 5.12. The MEKF bias estimate diverges significantly from the true value and lies well outside its predicted uncertainty bounds. This leads to substantial errors in the estimated angular velocity, which cascades into poor attitude propagation. By comparison, the LIEKF gyroscope bias estimate converges to the true value within 20 seconds, and the estimate error remains within its covariance bounds for the remainder of the simulation. This scenario highlights the limitations of the MEKF and demonstrates that the invariant structure of the LIEKF enables significantly improved estimation performance under large initialization errors and low measurement update rates.

Axis	MEKF	LIEKF
X	1.01141	0.00351
Y	0.99395	0.00829
Z	1.05972	0.00346

Table 5.6: RMS Attitude Estimation Error (radians) – MEKF Divergence Scenario

5.4 Monte Carlo Simulations

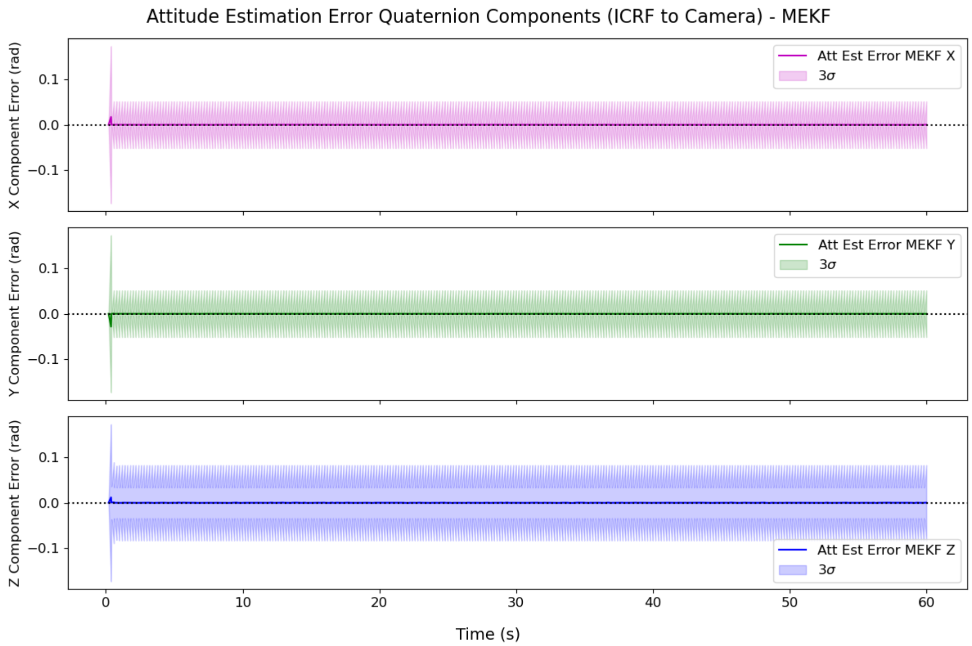
Section 5.3 demonstrates a single scenario in which the MEKF diverged. It is important to determine how often each filter diverges in order to accurately draw conclusions about their relative robustness. Thus, Monte Carlo simulations were conducted to evaluate how often each filter diverges under identical conditions. All trials used the system configuration described in Section 4.2.3, with the star tracker measurement sampling rate set to 1, 0.5, or 0.333 Hz. In this configuration, the initial attitude estimate for both filters is offset by 180° from the true attitude, and the standard deviation of the bias random walk is increased relative to the nominal case. For each sampling rate, 100 Monte Carlo trials were performed using the scenario defined in Section 4.1.4. Within each set, only the measurement noise realizations were varied.

Filter performance was evaluated using the convergence criteria defined in Section 4.3. The results are summarized in Table 5.7, which reports the number of successful trials for each filter at each sampling rate.

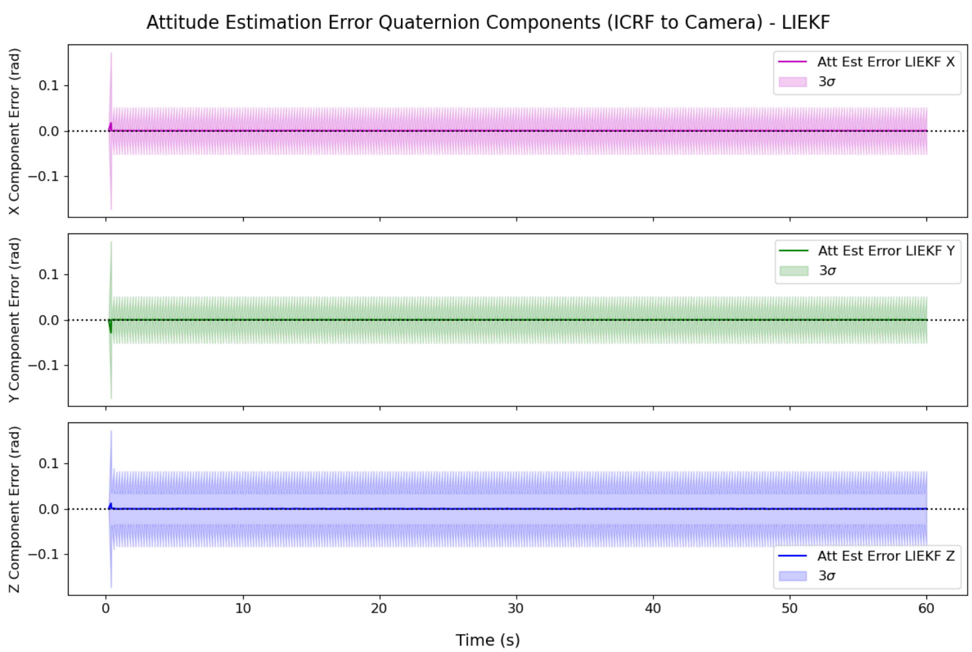
Star Tracker Sampling Rate (Hz)	1.0	0.5	0.33
MEKF	99	38	39
LIEKF	100	53	49

Table 5.7: Number of Converged Filters Out of 100 Monte Carlo Trials

Both filters exhibit near-perfect convergence when measurements are available at 1 Hz, indicating that frequent measurement updates mitigate the effects of poor initialization and nonlinear dynamics. As the measurement rate decreases, the likelihood of convergence of both filters also decreases. However, the LIEKF consistently outperforms the MEKF at lower sampling frequencies. Thus, we can make the following conclusion about the behavior of the filters: while both filters can perform well under favorable measurement conditions, the LIEKF exhibits greater robustness in scenarios characterized by large initial estimate errors and limited measurement availability.

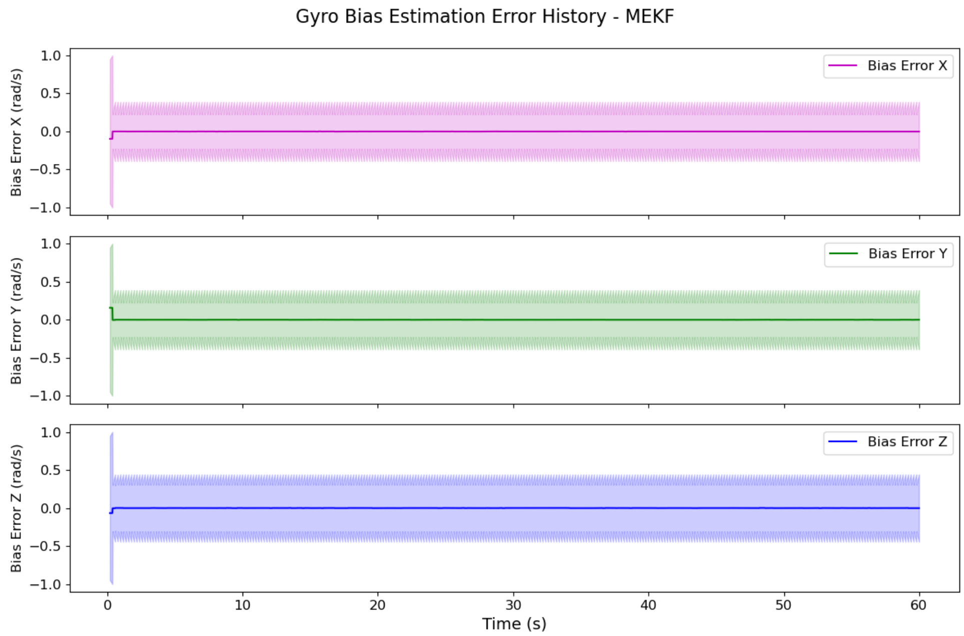


(a) MEKF

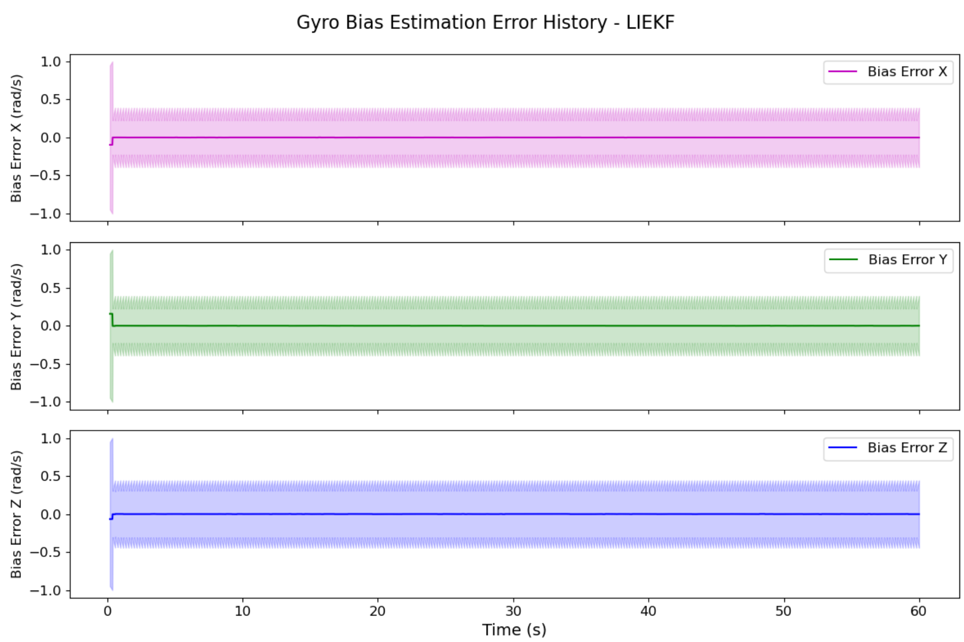


(b) LIEKF

Figure 5.1: Attitude Estimate Error – Inertial-Hold

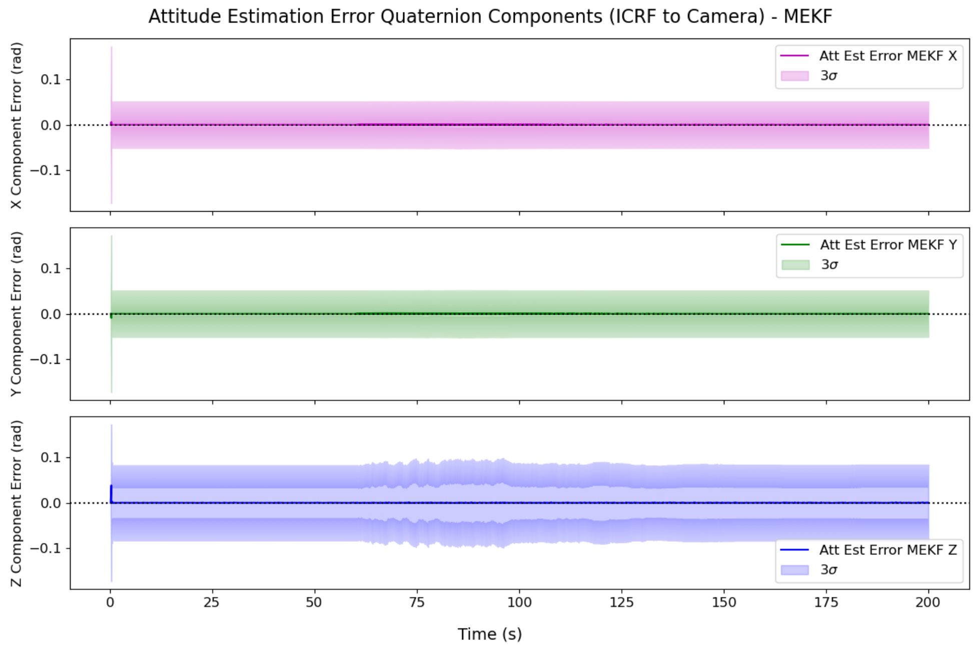


(a) MEKF

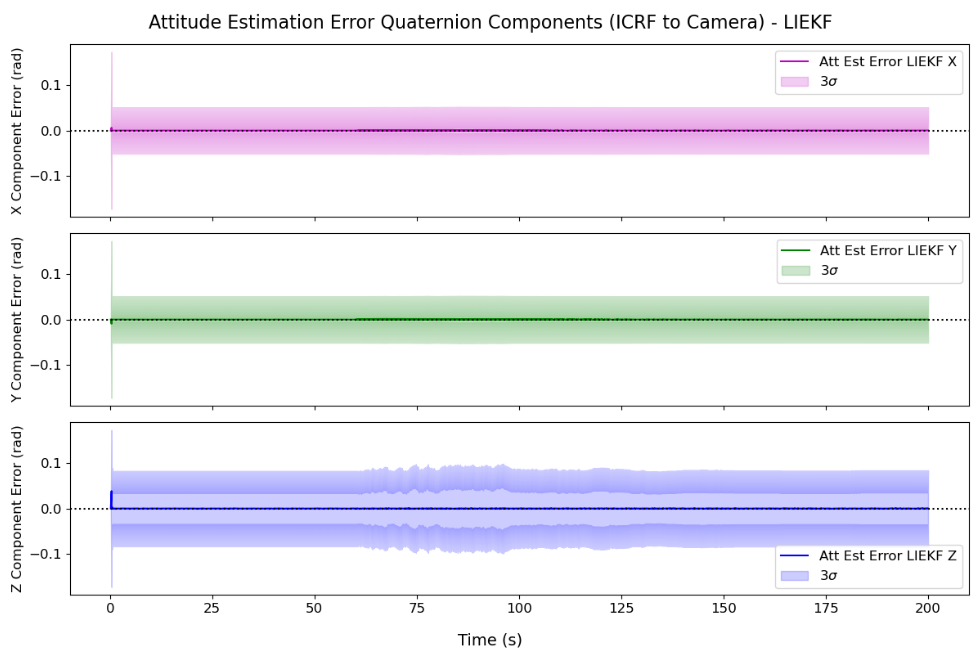


(b) LIEKF

Figure 5.2: Gyroscope Bias Estimate Error – Inertial-Hold

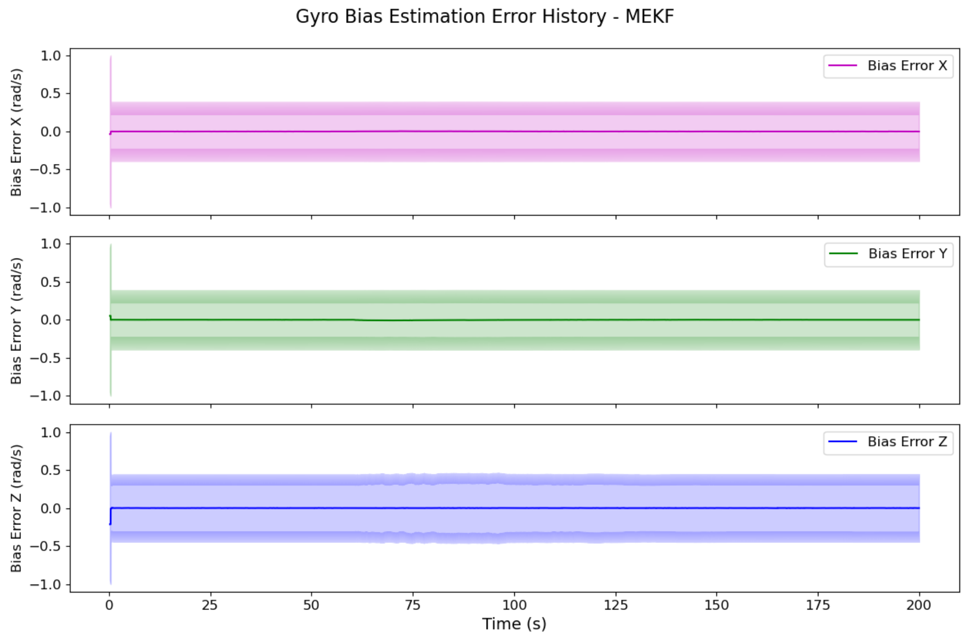


(a) MEKF

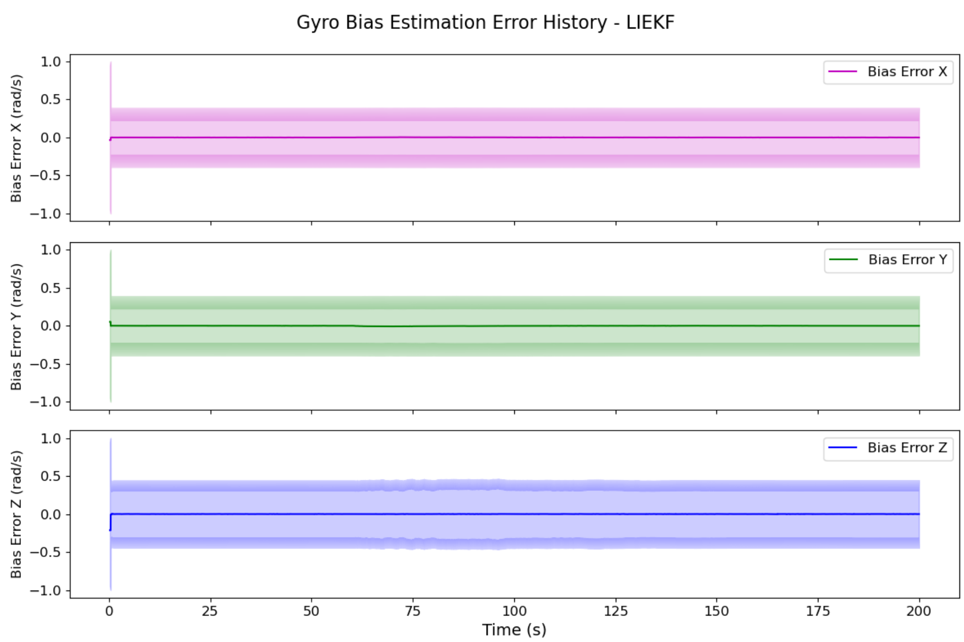


(b) LIEKF

Figure 5.3: Attitude Estimate Error – Slew

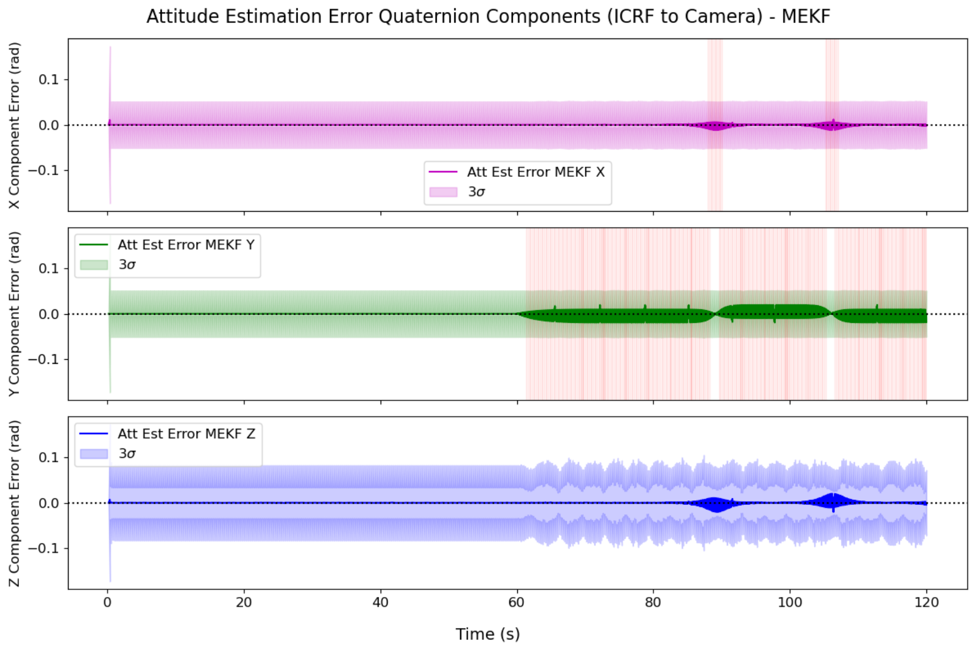


(a) MEKF

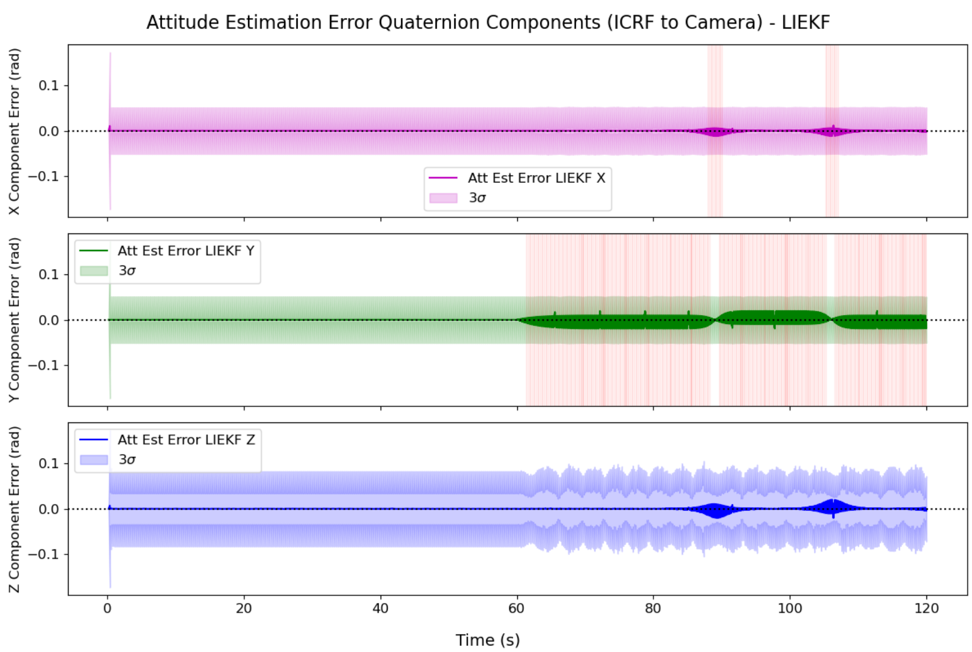


(b) LIEKF

Figure 5.4: Gyroscope Bias Estimate Error – Slew

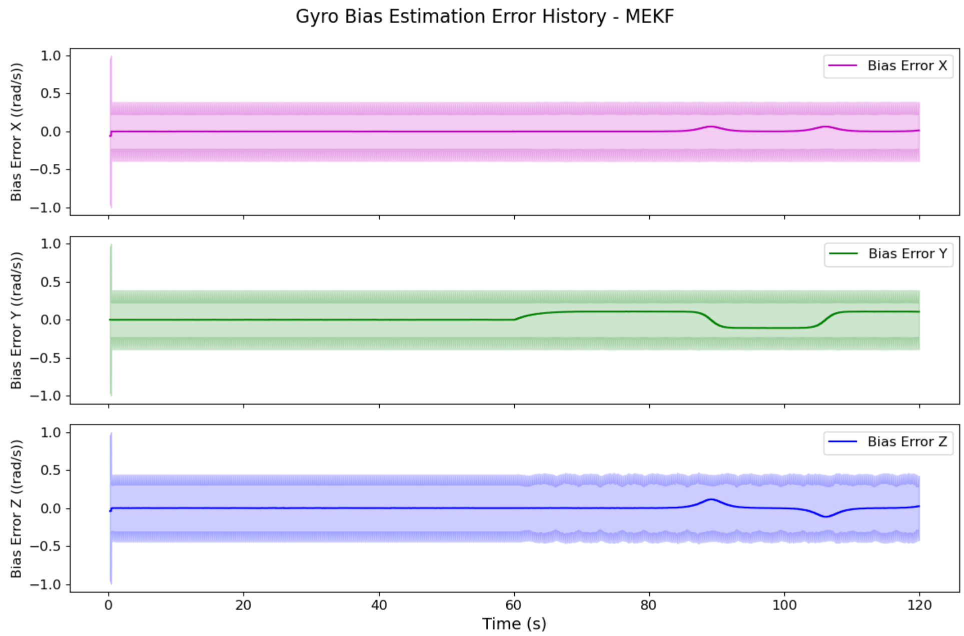


(a) MEKF

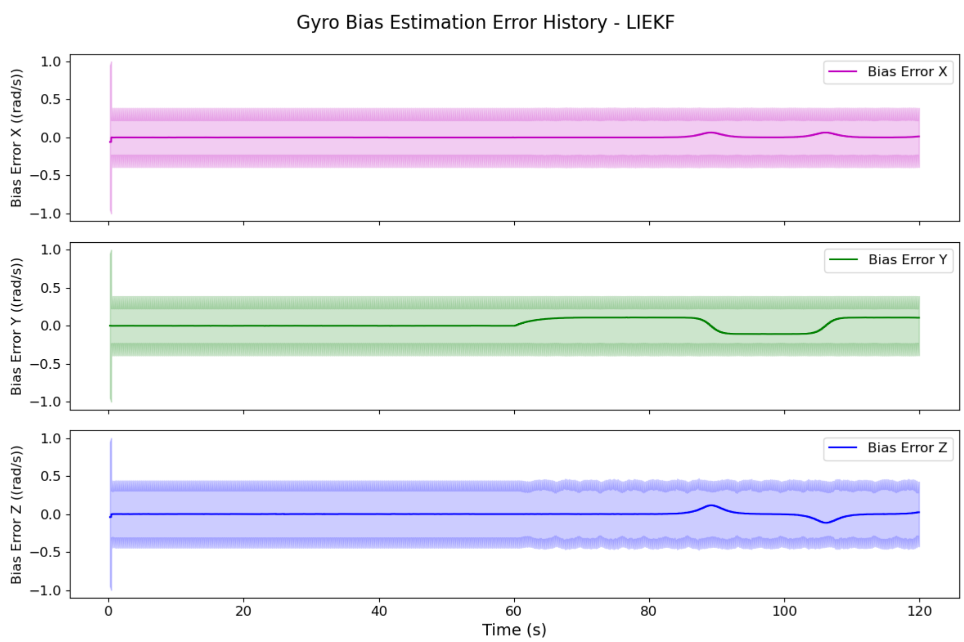


(b) LIEKF

Figure 5.5: Attitude Estimate Error – Intermediate Axis Spin & Tumble

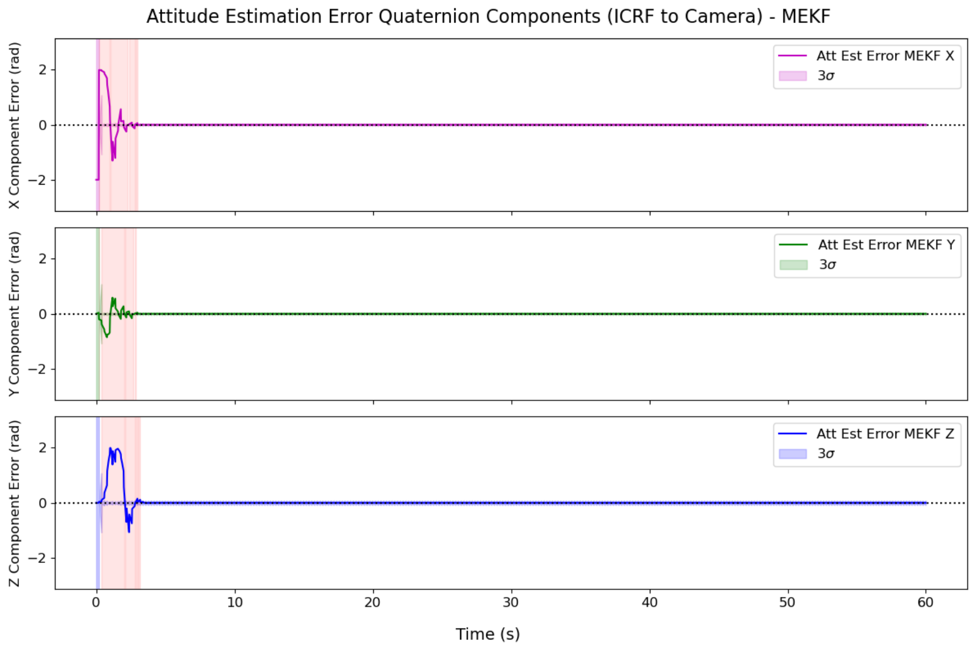


(a) MEKF

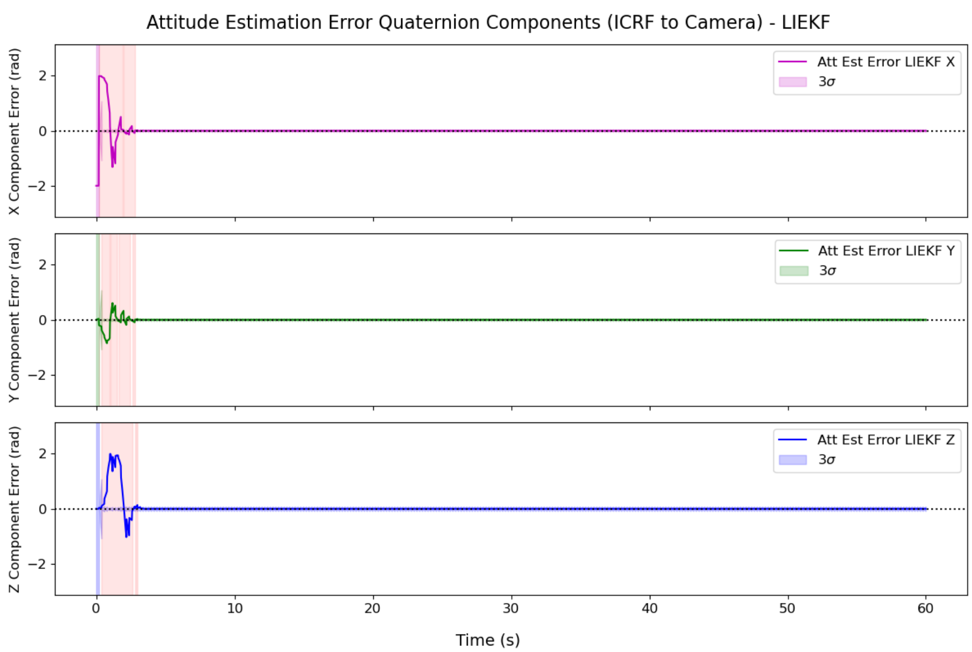


(b) LIEKF

Figure 5.6: Gyroscope Bias Estimate Error – Intermediate Axis Spin & Tumble

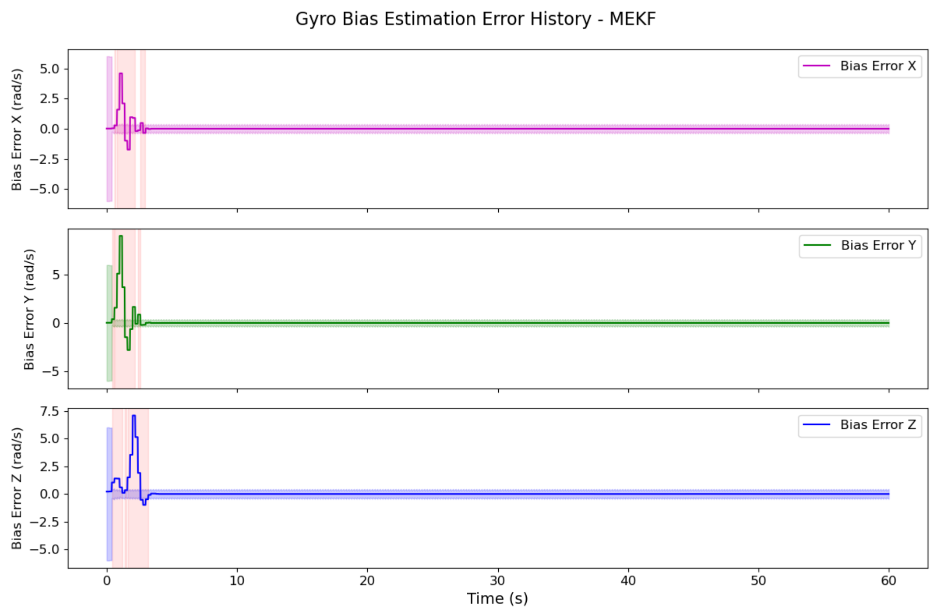


(a) MEKF



(b) LIEKF

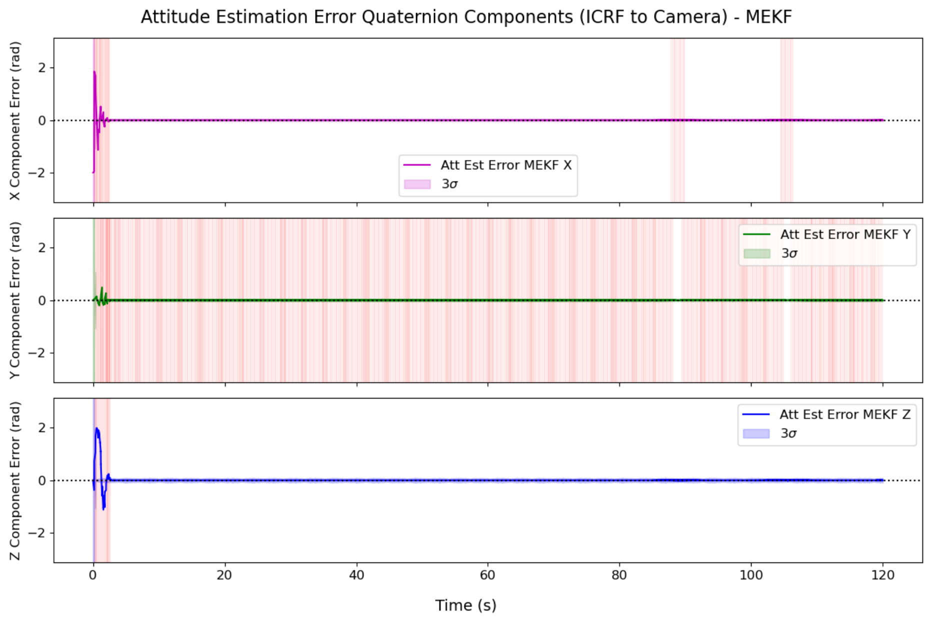
Figure 5.7: Attitude Estimate Error – Large Initial Attitude Estimate Error, Inertial-Hold



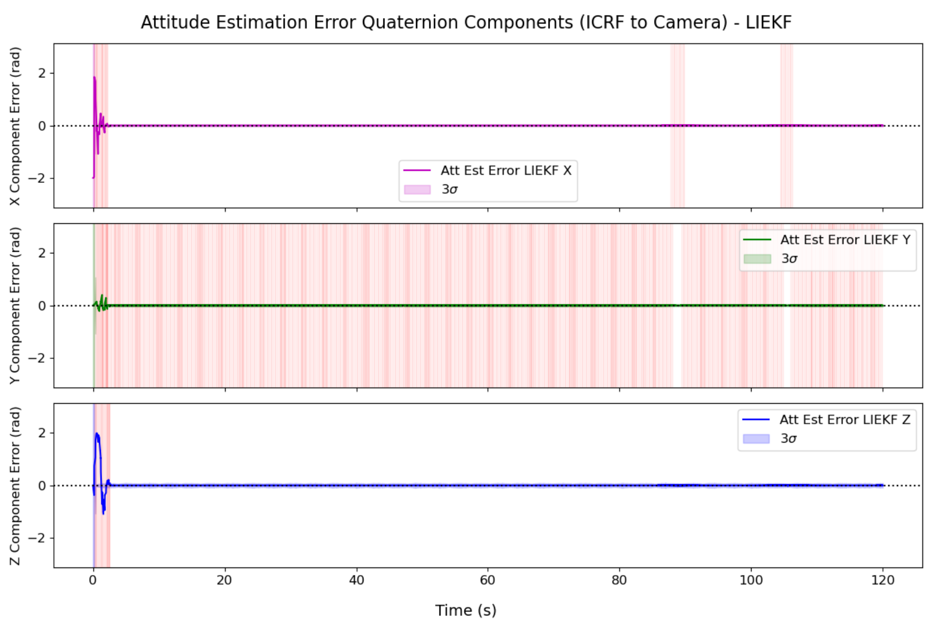
(a) MEKF

(b) LIEKF

Figure 5.8: Gyroscope Bias Estimate Error – Large Initial Attitude Estimate Error, Inertial-Hold

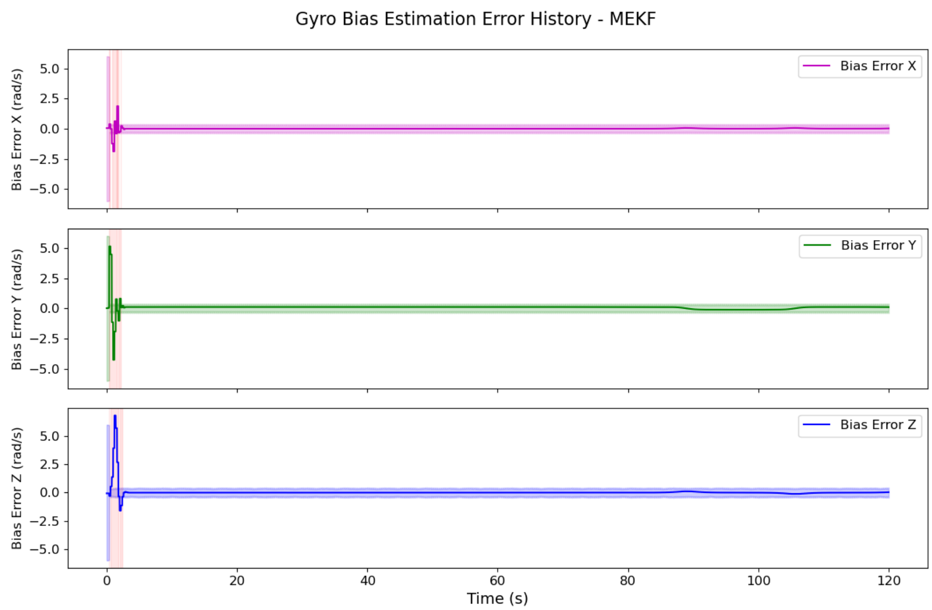


(a) MEKF

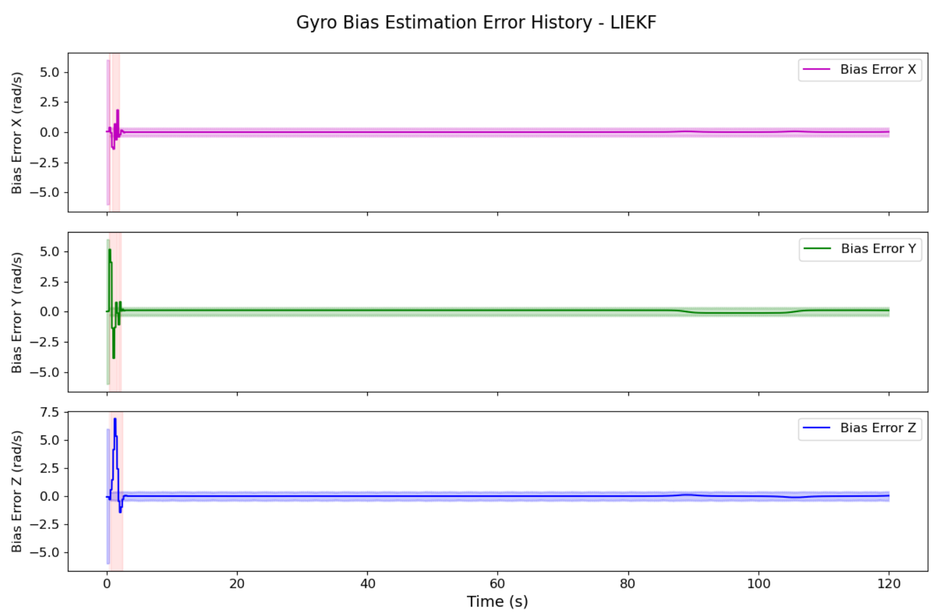


(b) LIEKF

Figure 5.9: Attitude Estimate Error – Large Initial Attitude Estimate Error, Intermediate Axis Tumble

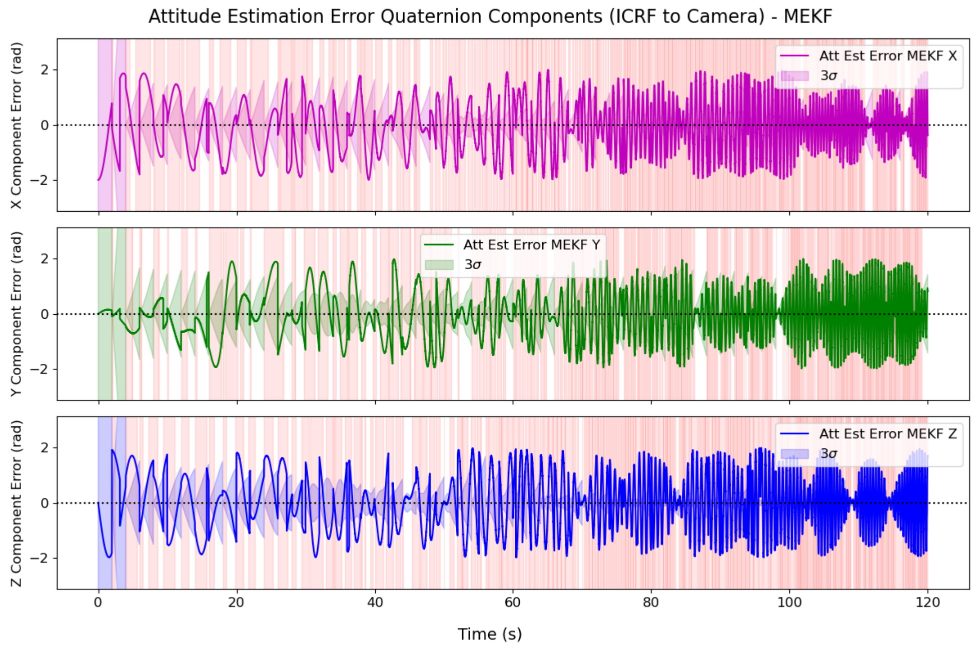


(a) MEKF

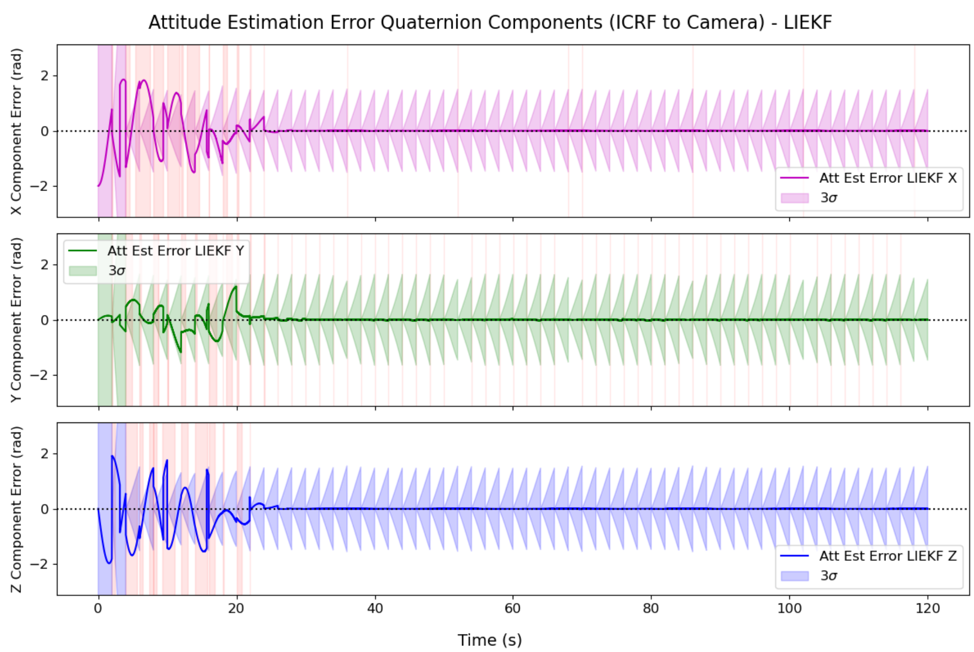


(b) LIEKF

Figure 5.10: Gyroscope Bias Estimate Error – Large Initial Attitude Estimate Error, Intermediate Axis Tumble

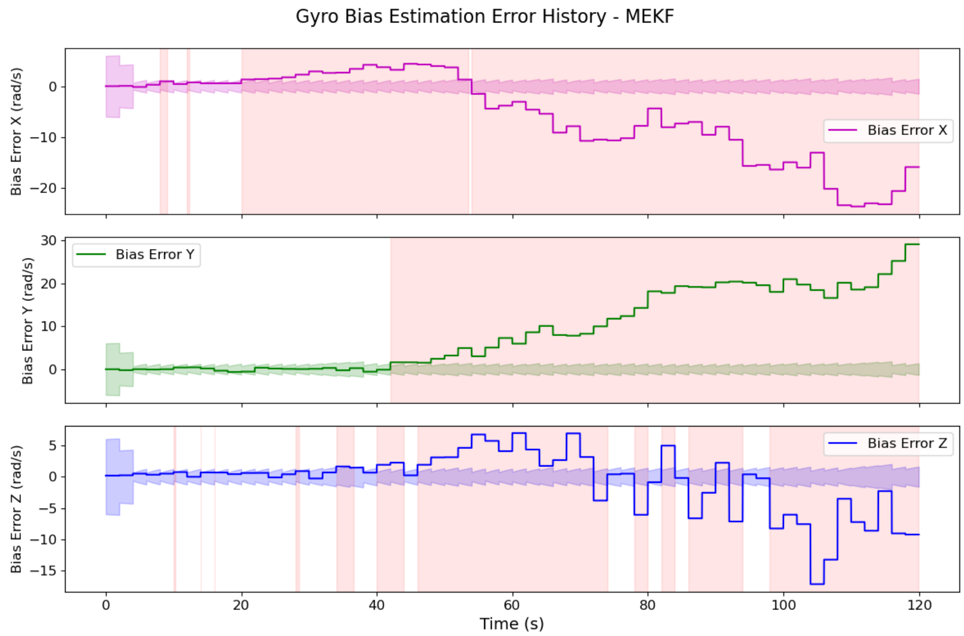


(a) MEKF

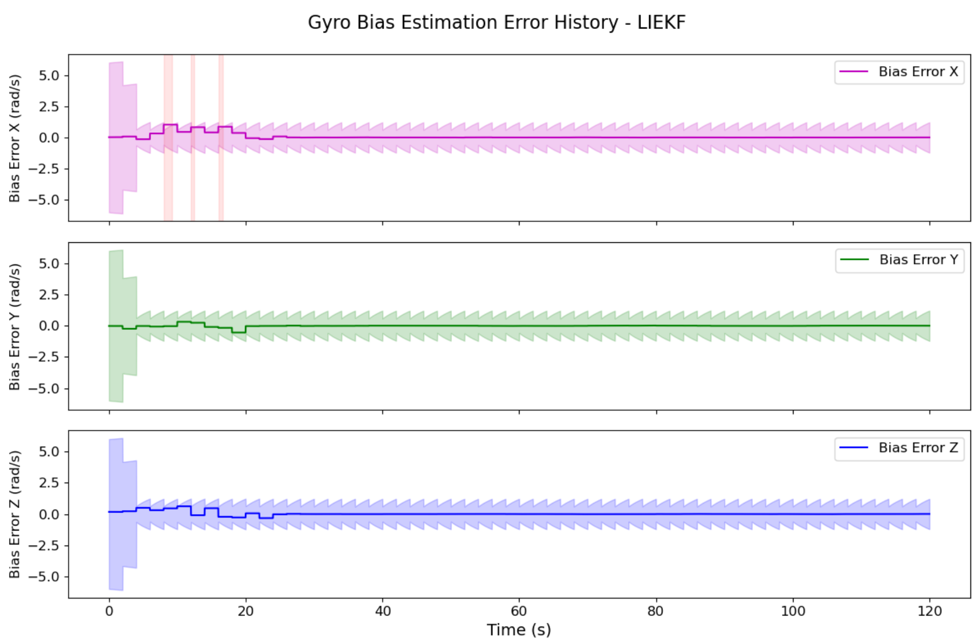


(b) LIEKF

Figure 5.11: Attitude Estimate Error – MEKF Divergence Scenario



(a) MEKF



(b) LIEKF

Figure 5.12: Gyroscope Bias Estimate Error – MEKF Divergence Scenario

Chapter 6
CONCLUSIONS

"What we know is a drop, what we don't know is an ocean."

attributed to Issac Newton

This thesis presents a simulation-based comparison of the Multiplicative Extended Kalman Filter (MEKF) and the Left-Invariant Extended Kalman Filter (LIEKF) applied to spacecraft attitude estimation using a rate-gyroscope and star tracker. A key result of this work is the demonstration that the spacecraft attitude estimation problem, formulated with a unit quaternion attitude state and rate-gyroscope bias state, satisfies the conditions required to apply the IEKF framework. Specifically, the state space of the system to be estimated forms a Lie group, the propagation of the error-state is state-trajectory independent, and the star tracker line-of-sight measurements are left-invariant. The resulting LIEKF inherits the local stability guarantees of the IEKF framework which the MEKF does not have. These theoretical guarantees motivate the expectation that the LIEKF should exhibit stronger convergence properties than the MEKF. Both filters are derived from a common error-state representation, with their algorithmic differences isolated to the quaternion update step.

In order to compare the performance of the filters, the system was evaluated under a series of dynamic scenarios and system configurations, ranging from nominal operating conditions with low initialization error and high-frequency measurements to highly nonlinear unstable dynamics with large initialization errors, more rate-gyroscope bias drift, and reduced measurement availability. Under nominal conditions, with low initialization error and high-frequency star tracker measurements, both filters performed comparably. In this configuration, the attitude and rate-gyroscope bias estimation errors remained low and well within the predicted uncertainty bounds for all dynamic scenarios evaluated, including attitude hold, slew maneuvers, and rotation about the intermediate principal axis. These results demonstrate that the LIEKF is a viable alternative to the MEKF under standard operating conditions as well as under unstable dynamics.

The primary advantage of the LIEKF was demonstrated under adverse conditions characterized by large initial attitude errors, increased gyroscope bias drift, unstable rotational dynamics, and reduced star tracker measurement availability. Under these conditions, the MEKF was shown to be susceptible to divergence, with unacceptably high attitude estimation errors and a gyroscope bias estimate that was inconsistent with its predicted uncertainty bounds. Under identical conditions and measurements, the LIEKF maintained a reliable state estimate, converging to the true attitude within approximately 20 seconds and remaining converged for the duration of the simulation. This result may be attributed to the LIEKF's local stability guarantees, which are unavailable to the MEKF.

The Monte Carlo analysis provided a statistical characterization of the robustness of the LIEKF. Across 100 trials at each of three star tracker measurement rates, the LIEKF converges in a greater proportion of trials than the MEKF. At a measurement rate of 1 Hz, both filters converged in nearly all trials, indicating that frequent measurements can mitigate the effects of poor initialization and nonlinear dynamics for both estimators. At reduced measurement rates of 0.5 and 0.33 Hz, the proportion of successful trials decreased for both filters, with the LIEKF successfully converging in significantly more trials than the MEKF.

These results demonstrate that the LIEKF is demonstrably more robust than the MEKF under the conditions most likely to cause estimator failure in practice. Given that the two filters differ only in the formulation of their quaternion update step and that the computation time of both filters update steps is approximately 0.0001 seconds, the adoption of the LIEKF imposes no additional computational burden over the MEKF. These results also suggest that by adopting the LIEKF, small satellite missions may be able to relax requirements on star tracker measurement rate or rate-gyroscope quality without sacrificing estimator reliability, potentially reducing the size, weight, power, and cost of the ADCS subsystem.

6.1 Future Work

The results and methodology presented in this work open several directions for future investigation, both in refining the evaluation of the LIEKF for spacecraft attitude estimation and in extending the IEKF framework to broader spacecraft navigation problems.

1. **Analysis with more restricted star-catalog** In this work, the simulated star tracker has access to a dense star catalog containing stars that may be too dim to be detected reliably, exhibit high proper motion, or have high parallax. In practice, flight star trackers use a reduced catalog to limit onboard memory requirements, increase identification robustness, and reduce computation time [2]. This reduces the number of stars available for attitude determination and can increase the frequency of periods during which too few stars are visible for a reliable attitude measurement. Evaluating both filters under these more realistic catalog constraints would provide a more accurate characterization of their relative performance in a flight-representative context.

2. **Analysis with star identification algorithms in the loop.** This work assumed perfect star identification, bypassing the matching process, and providing the filter with the true catalog identity of each observed star. A more realistic evaluation would incorporate a star identification algorithm directly into the simulation loop, introducing the possibility of incorrect or inconclusive identifications under noisy conditions. This would allow for a more realistic assessment of how measurement dropout and misidentification events affect the performance of both filters. Two scenarios of particular practical interest are the presence of other vehicles in the star tracker field of view, as occurs during rendezvous, proximity operations, and docking, and the presence of debris or particulate contamination from fluid venting, thruster plumes, or operation in debris-rich environments. In both cases, false star detections would degrade identification reliability and increase the frequency of measurement outages, directly testing the robustness advantage of the LIEKF identified in this work.

3. **Hardware-in-the-loop validation.** The results presented in this thesis are based entirely on simulation. While the simulation framework models realistic sensor noise characteristics and rigid-body dynamics, experimental validation on physical hardware is necessary to confirm that the performance advantages of the LIEKF observed in simulation translate to a real system. A hardware-in-the-loop testbed that incorporates a physical rate-gyroscope and a star tracker emulator would provide a more rigorous evaluation under conditions that are difficult to fully capture in simulation, such as the thermal effects of the space environment on gyroscope performance and the true noise characteristics of physical gyroscopes, which exhibit colored noise properties that are not fully captured by the white noise models used in simulation [28].

4. **Application to pose estimation.** Pose estimation, the joint estimation of attitude and position, is a natural extension of this work, particularly for applications such as in-space servicing, proximity operations, and formation flying, where precise knowledge of both orientation and relative position is required. The matrix Lie group $SE(3)$, which jointly encodes rotations and translations using dual quaternions, provides a state space for this problem within the IEKF framework. Previous works by Filipe, Kontitsis, and Tsiotras [33] have already applied the MEKF framework to the spacecraft pose estimation problem using dual-quaternions. Given the robustness advantages of the LIEKF demonstrated in this work, an IEKF-based pose estimator may offer similar theoretical and practical improvements over the dual-quaternion MEKF. This can be applied to relative navigation for rendezvous, proximity operations, and docking.

5. **Applications to Simultaneous Localization and Mapping.** Simultaneous localization and mapping (SLAM) is a process by which a vehicle can estimate its own state while building and maintaining a map of its environment. One such application to spacecraft navigation is a lunar lander that develops a map of the lunar surface from LiDAR measurements during descent and uses it to select a landing site [34]. A significant challenge in the development of such methods is the computational complexity

of jointly estimating the vehicle's state and a potentially large map. Recent work by Barrau and Bonnabel [19] has demonstrated that the IEKF yields provably consistent SLAM estimates under certain conditions where the standard EKF is known to be inconsistent, suggesting that an IEKF-based formulation could offer robustness advantages analogous to those observed in this work. This makes the IEKF a promising candidate for vision- or LiDAR-based terrain relative navigation for planetary landing and proximity operations around small bodies.

6. **Applications to Estimation on Constrained Manifolds.** In many estimation problems, the true state is known to lie on a constrained submanifold of the full state space, imposed by either physical requirements, operational constraints, or geometric relationships inherent to the system. Standard Kalman filter formulations do not natively enforce such constraints, potentially producing estimates or propagated uncertainties that violate them. Extending the IEKF framework to handle constrained manifolds using methods such as iSCVX [35] may provide improved estimator performance for systems with such constraints.

BIBLIOGRAPHY

[1] P. Charlot, C. S. Jacobs, D. Gordon, S. Lambert, A. de Witt, J. Böhm, A. L. Fey, R. Heinkelmann, E. Skurikhina, O. Titov, E. F. Arias, S. Bolotin, G. Bourda, C. Ma, Z. Malkin, A. Nothnagel, D. Mayer, D. S. MacMillan, T. Nilsson, and R. Gaume. The third realization of the International Celestial Reference Frame by very long baseline interferometry. *Astronomy & Astrophysics*, 644:A159, December 2020.

[2] F. Landis Markley and John L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Space Technology Library. Springer, New York, NY, 2014.

[3] R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *Journal of Basic Engineering*, 83(1):95–108, 03 1961.

[4] John L. Crassidis and John.L. Junkins. *Optimal Estimation of Dynamic Systems*. Chapman & Hall/CRC Applied Mathematics & Nonlinear Science. CRC Press, 2004.

[5] S. F. Schmidt. The kalman filter - its recognition and development for aerospace applications. *Journal of Guidance and Control*, 4(1):4–7, 1981.

[6] Malcolm D. Shuster. A survey of attitude representation. *Journal of The Astronautical Sciences*, 41:439–517, 1993.

[7] E.J. Lefferts, F.L. Markley, and M.D. Shuster. Kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, 5(5):417–429, 1982.

[8] John L. Crassidis, F. Landis Markley, and Yang Cheng. Survey of nonlinear attitude estimation methods. *Journal of Guidance, Control, and Dynamics*, 30(1):12–28, 2007.

[9] R.L. Farrenkopf. Analytic steady-state accuracy solutions for two common spacecraft attitude estimators. *Journal of guidance, control, and dynamics*, 1(4):282–284, 1978.

[10] Computer Sciences Corporation. Attitude Systems Operation, James Richard Wertz, and James R. Wertz. *Spacecraft attitude determination and control*, volume 73 of *Astrophysics and Space Science Library*. D. Reidel, Dordrecht, 1 edition, 1978.

[11] Clark E. Cohen, E. Glenn Lightsey, Bradford W. Parkinson, and William A. Feess. Space flight tests of attitude determination using gps. *International journal of satellite communications*, 12(5):427–433, 1994.

[12] James P. McQuerry, Richard A. Deters, and Mark A. Radovich, Jr. A precision star tracker for the nineties - A system guide to applications. In Robert D. Culp and Arlo D. Gravseth, editors, *IN: Guidance and control 1990; Proceedings of the Annual Rocky Mountain Guidance and Control Conference*, pages 83–104, January 1990.

[13] Sam Pedrotty, Ronney Lovelace, John Christian, Devin Renshaw, and Grace Quintero. Design and performance of an open-source star tracker algorithm on commercial off-the-shelf cameras and computers. In *Advances in the Astronautical Sciences AAS Guidance, Navigation, and Control 2020*, volume 172, February 2020.

[14] C.C. Liebe. Accuracy performance of star trackers - a tutorial. *IEEE transactions on aerospace and electronic systems*, 38(2):587–599, 2002.

[15] Craig L. Cole and John L. Crassidis. Fast star-pattern recognition using planar triangles. *Journal of Guidance, Control, and Dynamics*, 29(1):64–71, 2006.

[16] John A. Christian and John L. Crassidis. Star identification and attitude determination with projective cameras. *IEEE Access*, 9:25768–25794, 2021.

[17] C.C. Liebe. Pattern recognition of star constellations for spacecraft applications, 1992.

[18] Paul McKee, Jacob Kowalski, and John A. Christian. Navigation and star identification for an interstellar mission. *Acta Astronautica*, 192:390–401, 2022.

[19] Axel Barrau and Silvère Bonnabel. Invariant Kalman filtering. *Annual Review of Control, Robotics, and Autonomous Systems* , 1(1):237–257, May 2018.

[20] Axel Barrau and Silvère Bonnabel. The invariant extended kalman filter as a stable observer. *IEEE Transactions on Automatic Control*, 62(4):1797–1812, 2017.

[21] Axel Barrau and Silvere Bonnabel. Intrinsic filtering on lie groups with applications to attitude estimation. *IEEE transactions on automatic control*, 60(2):436–449, 2015.

[22] Silvere Bonnabel. Left-invariant extended kalman filter and attitude estimation. In *2007 46th IEEE Conference on Decision and Control*, pages 1027–1032, 2007.

[23] Haichao Gui and Anton H. J. de Ruiter. Quaternion invariant extended kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, 41(4):863–878, 2018.

[24] Brian C. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*. Springer, New York, NY, 2003.

[25] ESA, editor. *The HIPPARCOS and TYCHO catalogues. Astrometric and photometric star catalogues derived from the ESA HIPPARCOS Space Astrometry Mission*, volume 1200 of *ESA Special Publication*, January 1997.

[26] Yi Ma. An invitation to 3-d vision : from images to geometric models, 2004.

[27] John Christian, Lylia Benhacine, Jacob Hikes, and Christopher D'Souza. Geometric calibration of the orion optical navigation camera using star field images. *The Journal of the Astronautical Sciences*, 63, 07 2016.

[28] Institute of Electrical and Electronics Engineers. Ieee standard specification format guide and test procedure for single-axis laser gyros. *IEEE Std 647-2006 (Revision of IEEE Std 647-1995)*, pages 1–96, 2006.

[29] Grace Wahba. A least squares estimate of satellite attitude. *SIAM review*, 7(3):409–409, 1965.

[30] Paul B. Davenport. A vector approach to the algebra of rotations with applications. Technical report, National Aeronautics and Space Administration, 08 1968.

Available at: https://ntrs.nasa.gov/api/citations/19680021122/downloads/19680021122.pdf.

[31] Malcom D. Shuster. Kalman filtering of spacecraft attitude and the quest model. *The Journal of the Astronautical Sciences*, 38(3):377–393, 1990.

[32] Parth S. Shrotri. Invariant extended kalman filter for spacecraft attitude determination. Available at: https://github.com/parthshrotri/IEKF_Attitude_Determination, 2026.

[33] Nuno Filipe, Michail Kontitsis, and Panagiotis Tsiotras. Extended kalman filter for spacecraft pose estimation using dual quaternions. In *2015 American Control Conference (ACC)*, pages 3187–3192, 2015.

[34] Kevin Kobylka, Davis Adams, M. Marshall, Parth Shrotri, and Benjamin Benjadol. Standalone hazard evaluation and refinement from instrument findings (s.h.e.r.i.f.). In *47th Annual AAS Guidance, Navigation & Control Conference Proceedings*, Johnson Space Center, 2025.

[35] Spencer Kraisler, Mehran Mesbahi, and Behçet Açıkmeşe. Intrinsic successive convexification: Trajectory optimization on smooth manifolds. *IEEE Control Systems Letters*, 9:408–413, 2025.

Appendix A
SYSTEM ARCHITECTURE

Table A.1 summarizes the configuration structure used to define each simulation case. The simulation configuration is divided into scenario-level parameters which describe the events in the simulation and system-level parameters which describe the properties of the vehicle, its sensors, and flight software configuration. This enables the evaluation of different scenarios and system characteristics without modifying the core simulation architecture.

Figure A.1 is an illustration of the high-level simulation architecture. The system configuration files are provided to the simulation core, which instantiates the vehicle, its sensors, and flight software according to the specified parameters. The vehicle object maintains the ground truth of the system, including its true attitude, angular velocity, position, velocity, and propagates states according to the control inputs and dynamics of the system. Sensors are implemented as attributes of the vehicle and generate measurements based on the true state and incorporate noise based on the characteristics defined in the system configuration

The simulation core provides a unified time base and coordinates the interaction between the vehicle and the flight software. The simulation core parses user-generated commands and measurements generated by the sensors and passes them to the flight software. It also processes and sends commands from the flight software to the vehicle, such as modifications to the commanded torque.

Figure A.2 illustrates the implementation of the flight software in the simulation environment. Attitude commands and sensor measurements are received by the flight software and routed to the attitude controller and attitude estimators, respectively. Two attitude estimators, the Multiplicative Extended Kalman Filter (MEKF) and the Left-Invariant Extended Kalman Filter (LIEKF), execute in parallel to produce independent estimates of the attitude and rate-gyroscope bias. Both estimators process the same measurements provided by the rate-gyroscope and star tracker. In a flight-representative implementation, the attitude controller would utilize estimated states. However, in this simulation, the attitude controller is driven by the ground truth attitude. This is done to decouple the estimation and control schemes to enable independent analysis of estimator performance. Attitude control torque commands are subsequently sent by the flight software to the vehicle's actuators.

The attitude control system supports three different modes of operation: Direct Torque Control, Rate, and Inertial Attitude Pointing. These commands are grouped under the `GNC_ATT_CMD` identifier to indicate that they modify the operating mode or setpoints of the attitude control system. Table A.2 summarizes the set of commands used to transition between these modes.

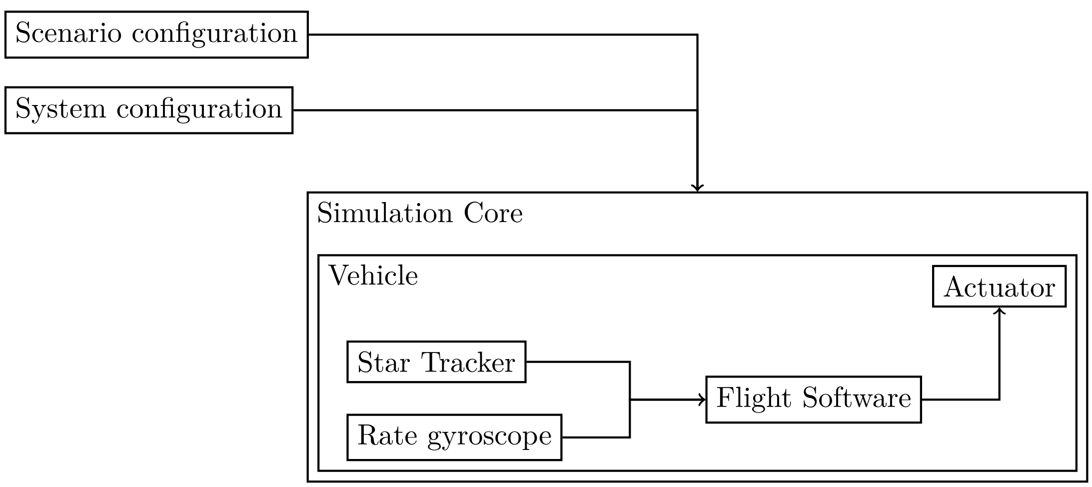


Figure A.1: Simulation Architecture

Scenario Configuration Parameters
• Input and Output files • Start time, Duration, Time step Size • Initial conditions • Attitude commands • Number of Monte Carlo Runs (Optional)
System Configuration Parameters
• Vehicle identifier and input file names • Vehicle mass and inertia properties • Flight software and sensor Sample Rates • Gyro noise and bias parameters • Star catalog • Star tracker camera intrinsic parameters • Star tracker centroiding standard deviation • Attitude controller gains • Kalman filter initial state estimate • Kalman filter process and measurement noise parameters

Table A.1: Configuration Parameters

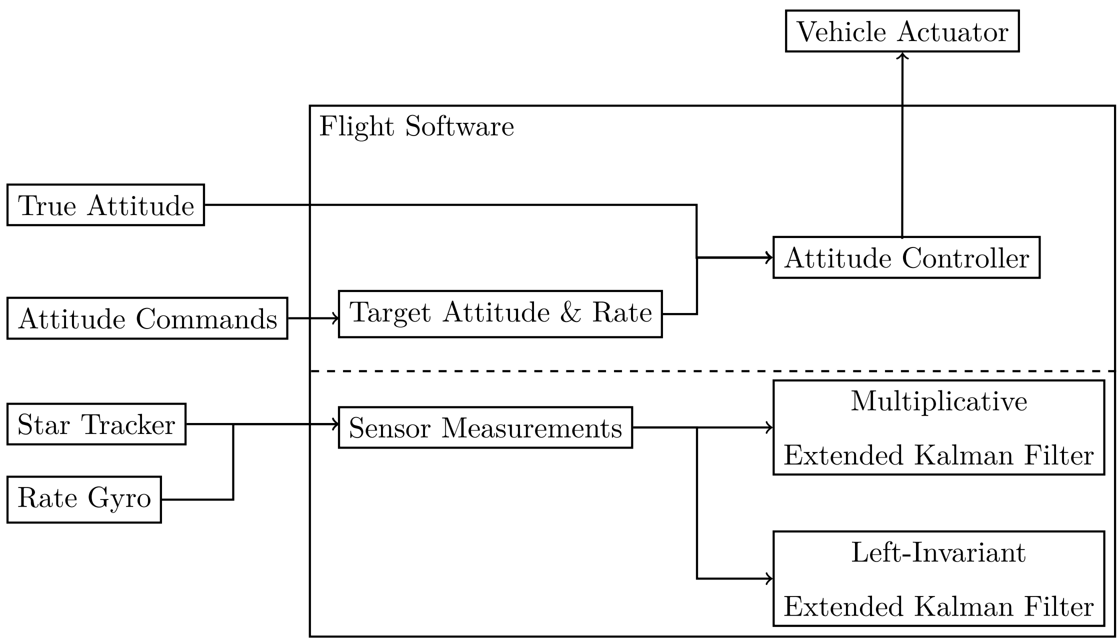


Figure A.2: Flight Software Architecture

Direct Torque Control Mode	
"TORQUE(τ_x, τ_y, τ_z)"	Bypass attitude controller, sets actuator torque vector to (τ_x, τ_y, τ_z) Nm
"OFF"	Bypass attitude controller, Removes attitude and angular rate setpoints, sets actuator torque vector to (0, 0, 0) Nm
"AUTO_TORQUE"	Determine actuator torque based on attitude controller (used to restore attitude control authority after it has been bypassed by "TORQUE" or disabled via "OFF")
Rate Mode	
"RATE(ω_x,ω_y,ω_z)"	Removes inertial-to-body target, sets body-frame angular rate setpoint to $(\omega_x, \omega_y, \omega_z)$ rad/sec
Inertial Attitude Pointing Mode	
"Q(q_x,q_y,q_z,q_s)"	Updates inertial-to-body target to the specified scalar-last quaternion and sets body-frame angular velocity setpoint to (0,0,0) rad/sec
"RA(α),DEC(δ)"	Calculates and sets target inertial-to-body quaternion based on specified Right-Ascension and Declination (α and δ in degrees) and sets body-frame angular velocity setpoint to (0,0,0) rad/sec
"HIP($n_{\mathtt{HIP}}$)"	Calculates and sets target inertial-to-body quaternion based on specified Hipparcos star catalog ID number (n_{HIP}) and sets body-frame angular velocity setpoint to (0,0,0) rad/sec
"SPICE($n_{\mathtt{SPICE}}$)"	Calculates and sets target inertial-to-body quaternion based on SPICE target ID number (n_{SPICE}) sets body-frame angular velocity setpoint to (0,0,0) rad/sec

Table A.2: Attitude Control System Mode Commands

Appendix B

DEFAULT SIMULATION CONFIGURATION

This appendix provides a representative example of a scenario and system configuration used in the trials. The other evaluated configurations follow the same structure and differ only in the parameter variations given in Sections 4.1 and 4.2. In the simulation code-base, these tables are given in YAML files.

Input Files	
SPICE kernel directory location	`input/kernels/`
Required SPICE Kernels	• de440.bsp • naif0012.tls
System configuration directory location	`sim_configs/default/`
Output Files	
Output file directory location	`"outputs/nominal_static/"`
Simulation Parameters	
Start time	2005-08-13T00:00:00.000
Duration	60 sec
Time step size	10 millisec
Initial attitude quaternion	[0,0,0,1]
Initial angular velocity	(0.0,0.0,0.0) rad/sec
Number of Monte Carlo Runs (Optional)	–

Commands to send to flight software
• Seconds elapsed since start
• Command type
• Command
	
	`time: 0,`
	`type: "GNC_ATT_CMD"`
	`cmd: "Q(0,0,0,1)"`

Table B.1: Default Scenario Configuration Parameters

Spacecraft Name	`"Spacecraft1"`
SPICE_ID	-74
SPICE Kernel	`"mro_cruise.bsp"`
Inertia Tensor	diag([15000, 10000, 4000]) kg/m^2
Star Tracker Configuration Relative Path	`sensors/camera.yaml`
Vehicle Body to Star Tracker Frame Quaternion	[0,0,0,1]
Rate Gyro Configuration Relative Path	`sensors/rategyro.yaml`

Table B.2: Default Vehicle Configuration Parameters

Focal Length / Pixel Width	7310 pixels
Focal Length / Pixel Height	7310 pixels
Skewness	0
Principal Point (x)	1024 pixels
Principal Point (y)	1024 pixels
Number of Pixel Rows	2048 pixels
Number of Pixel Columns	2048 pixels
Half Angle Field of View	10 degrees
Camera Boresight Vector	[0, 0, 1]
Line-of-Sight Vector Standard Deviation	1e-4 radians
Star Catalog Relative Path	`inputs/catalog/hip.csv`
Star Catalog Epoch	J2000
Maximum Apparent Visual Magnitude	6.5
Planets to Include in Star Tracker Catalog	[1, 2, 3, 4, 5, 6]
Save Generated Star Field Images	True
Sample Rate	5 Hz

Table B.3: Default Star Tracker Configuration Parameters

Sample Rate	50 Hz
Noise Standard Deviatiation	0.001 rad/sec
Initial Bias Standard Deviation	0.1 rad/sec
Bias Rate Standard Deviation	0.001 rad/sec^2

Table B.4: Default Rate-Gyroscope Configuration Parameters

Sample Rate	50 Hz
Attitude Controller Proportional Gains	[400, 360, 480]
Attitude Controller Derivative Gains	[4000, 3600, 4800]
Filter Initial Estimate	`"AUTO"` (Davenport's q Method)
Filter Initial Bias Standard Deviation	0.1 rad/sec
Filter Gyro Noise Standard Deviation	0.01 rad/sec
Filter Gyro Bias Rate Standard Deviation	0.25 rad/sec^2
Filter Star Tracker Line-of-Sight Standard Deviation	0.01 radians
Star Tracker Catalog Relative Path	`inputs/catalog/hip.csv`
Star Tracker Catalog Epoch	J2000
Maximum Apparent Visual Magnitude	6.5

Table B.5: Default Flight Software Configuration Parameters