A COMPARISON STUDY OF STATE ESTIMATORS FOR A SPHERICAL PENDULUM

A THESIS SUBMITTED TO THE GRADUATE DIVISION OF THE UNIVERSITY OF HAWAI’I AT MĀNOA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

MECHANICAL ENGINEERING

MAY 2011

By

ELIZABETH DIMMITT GREGORY

Thesis Committee:
Amit K. Sanyal, Chairperson
Trevor Sorensen
Brian Bingham
Luke Flynn
We certify that we have read this thesis and that, in our opinion, it is satisfactory in scope and quality as a thesis for the degree of Master of Science in Mechanical Engineering.

THESIS COMMITTEE

______________________________
Chairperson

______________________________

______________________________

______________________________
Copyright 2011 by
ELIZABETH DIMMITT GREGORY
To my Mother and Father

who have always loved and supported me
and taught me how to do what is right and good.

To Dr. Trevor Sorensen

who brought me to Hawaii in the first place.
ACKNOWLEDGMENTS

I want to thank my advisor Dr. Amit Sanyal who gave me this wonderful work to do and helped me understand it.

I want to thank Dr. Nikolaj Nordkvist who was kind enough to put up with lots of questions and is one of the best teachers I have ever had.

I want to thank Zachary Lee-Ho, Miguel Nunes, and Jeremy Chan for all their help and listening to me talk about this project. I also need to thank the rest of the HSFL staff for giving me time to work on this.
ABSTRACT

In the field of spacecraft attitude determination and control, the prevailing state estimation method is the Extended Kalman Filter (EKF). Often, with the EKF, a lower dimensional representation is used to describe the attitude of the spacecraft which leads to singularities and ambiguities. The EKF also requires the definition of the uncertainty of the process and sensor noise to be described with a Gaussian distribution. Hawai‘iSat-1 plans to use an alternative, deterministic estimation scheme which uses the global representation of attitude and angular velocity and which only requires the measurement noise and the initial state uncertainty to be bounded by known ellipsoidal bounds. These bounds are referred to as uncertainty ellipsoids.

In order to reduce risk associated with using an estimation scheme with no flight history, a comparison of this deterministic estimation scheme with an EKF was conducted for a spherical pendulum as a simplified analogy for the pointing needs of a satellite. This thesis documents the comparison.

The deterministic estimation schemes use a reduced attitude vector and the angular velocity vector to represent the state. This state representation is also used to define uncertainty ellipsoids characterizing bounds on the state estimate error and sensor measurement error. The Extended Kalman Filter uses spherical coordinates to represent the state and Gaussian noise to describe uncertainty of both the estimate and sensor readings.

The comparison concludes with the deterministic estimator performing comparably or better than the EKF for various initial conditions and noise models.

Keywords: Spacecraft Attitude Estimation, Deterministic State Estimation, and Kalman Filtering
# TABLE OF CONTENTS

**Acknowledgments** .................................................................................. v

**Abstract** ................................................................................................... vi

**List of Tables** ............................................................................................. ix

**List of Figures** ........................................................................................... x

1 **Introduction** ........................................................................................... 1

2 **Spherical Pendulum System** .................................................................. 4
   2.1 Representations ..................................................................................... 4
   2.2 Physical Description .............................................................................. 7
   2.3 Derivation of Continuous Equations of Motion .................................... 8
      2.3.1 Global Representation .................................................................. 8
      2.3.2 Spherical Coordinates ................................................................... 9

3 **Estimation Schemes** ............................................................................... 12
   3.1 Deterministic Estimation Scheme .......................................................... 12
      3.1.1 Uncertainty Ellipsoids .................................................................... 13
      3.1.2 Propagation of State Estimates Using a Variational Integrator ....... 14
      3.1.3 Propagation of Uncertainty Bounds on State Estimates ................. 16
      3.1.4 Filtering Update .......................................................................... 20
   3.2 Extended Kalman Filter ....................................................................... 23
      3.2.1 Covariance Matrix ....................................................................... 23
      3.2.2 Propagation ................................................................................ 23
      3.2.3 Update ...................................................................................... 24
LIST OF TABLES

5.1 Initial States ................................................................. 37
5.2 Noise Model Definition ..................................................... 38
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Inertial and Body Reference Frames</td>
<td>5</td>
</tr>
<tr>
<td>2.2</td>
<td>Spherical Coordinates</td>
<td>7</td>
</tr>
<tr>
<td>3.1</td>
<td>Process of Ellipsoid Propagation and Intersection</td>
<td>21</td>
</tr>
<tr>
<td>4.1</td>
<td>Conversion Error</td>
<td>33</td>
</tr>
<tr>
<td>4.2</td>
<td>Linearization Test of $A_k$</td>
<td>35</td>
</tr>
<tr>
<td>4.3</td>
<td>System Chaos Test</td>
<td>36</td>
</tr>
<tr>
<td>5.1</td>
<td>Pendulum in Hanging Position</td>
<td>37</td>
</tr>
<tr>
<td>5.2</td>
<td>Pendulum in Inverted Position</td>
<td>38</td>
</tr>
<tr>
<td>5.3</td>
<td>Noise Distributions for simulation</td>
<td>39</td>
</tr>
<tr>
<td>5.4</td>
<td>Global Representation Components for EKF for Inverted Initial State and Gaussian Noise</td>
<td>40</td>
</tr>
<tr>
<td>5.5</td>
<td>Global Representation Components for D.E. for Inverted Initial State and Gaussian Noise</td>
<td>40</td>
</tr>
<tr>
<td>5.6</td>
<td>Global Representation Components for EKF for Inverted Initial State and Gaussian Noise (Second Simulation)</td>
<td>41</td>
</tr>
<tr>
<td>5.7</td>
<td>Global Representation Components for D.E. for Inverted Initial State and Gaussian Noise (Second Simulation)</td>
<td>41</td>
</tr>
<tr>
<td>5.8</td>
<td>Error for the Inverted Initial Condition and Gaussian Noise</td>
<td>42</td>
</tr>
<tr>
<td>5.9</td>
<td>Uncertainty for the Inverted Initial Condition and Gaussian Noise</td>
<td>42</td>
</tr>
<tr>
<td>5.10</td>
<td>Error for the Inverted Initial Condition and Gaussian Noise (Second Simulation)</td>
<td>43</td>
</tr>
</tbody>
</table>
5.11 Uncertainty for the Inverted Initial Condition and Gaussian Noise (Second Simulation) ...................................................... 43
5.12 Global Representation Components for EKF for Hanging Initial State and Gaussian Noise .................................................. 44
5.13 Global Representation Components for D.E. for Hanging Initial State and Gaussian Noise .................................................... 44
5.14 Error for the Hanging Initial Condition and Gaussian Noise .......... 45
5.15 Error for the Hanging Initial Condition and Gaussian Noise .......... 45
5.16 Noise Distributions for simulation ............................................. 46
5.17 Global Representation Components for EKF for Inverted Initial State for Uniformly Distributed Noise .................................. 47
5.18 Global Representation Components for D.E. for Inverted Initial State for Uniformly Distributed Noise .................................. 47
5.19 Global Representation Components for EKF for Inverted Initial State for Uniformly Distributed Noise (Second Simulation) .......... 48
5.20 Global Representation Components for D.E. for Inverted Initial State for Uniformly Distributed Noise (Second Simulation) .......... 48
5.21 Error for the Inverted Initial Condition Using Uniformly Distributed Noise ................................................................. 49
5.22 Uncertainty for the Inverted Initial Condition Using Uniform Distributed Noise ............................................................. 49
5.23 Error for the Inverted Initial Condition Using Uniformly Distributed Noise (Second Simulation) ................................................. 50
5.24 Uncertainty for the Inverted Initial Condition Using Uniform Distributed Noise (Second Simulation) ........................................... 50
5.25 Global Representation Components for EKF for Hanging Initial State for Uniformly Distributed Noise .................................. 51
5.26 Global Representation Components for D.E. for Hanging Initial State for Uniformly Distributed Noise .................................. 51
5.27 Error for the Hanging Initial Condition Using Uniformly Distributed Noise ................................................................. 52
5.28 Uncertainty for the Hanging Initial Condition Using Uniform Distributed Noise
CHAPTER 1
INTRODUCTION

State estimation is a complicated problem that has applications in a wide range of research fields including GPS, robotics, auto-pilot systems, tracking applications, and spacecraft attitude estimation. State estimation is needed for any application where the state is evolving with time and there is uncertainty associated with both the dynamics process model and any measurements of the state.

In the field of spacecraft attitude determination and control, the state consists of a representation of the orientation of the spacecraft body frame with respect to an inertial coordinate frame and a representation of the rate of change of the attitude. By far the most popular method for state estimation in the field of spacecraft attitude estimation is the Extended Kalman Filter and its derivatives[1]. The challenge for state estimation in this field has in some ways become one of attitude representation instead of estimation. As Crassidis, Markley and Cheng point out in Reference [1], most EKF’s do not use the nine parameter, rotation matrix to represent the three dimensional problem of rotation (attitude) but instead use a coordinate representation with fewer parameters. The most common coordinate systems that are used to represent spacecraft attitude include quaternions, Euler angles and Rodrigues parameters. The ideal application for an EKF is one where a three dimensional problem is represented with three parameters so to facilitate a linear application of the Kalman gain. However, all representations short of the global rotation matrix result in points of ambiguity and discontinuity where the parameters cannot fully or uniquely define the attitude.

The Kalman Filter was originally proposed by Kalman in 1960 as the solution to the optimal estimation problem for linear systems with model and sensor noise described with a Gaussian distribution[2]. The Extended Kalman Filter (EKF) applies the same procedure to nonlinear systems by linearizing the model about the estimation point at each time step. The EKF is not an optimal solution but has proven to be a “good” solution and has
been applied to attitude estimation since 1970 [3]. The process of the Kalman filter update step includes linearly applying a Kalman gain that is calculated based on the ratio of the uncertainties of the model and the uncertainty of the sensor. This process works with coordinates, but does not work well with a global representation of the state space where the state variables are not independent of each other.

These discontinuities have led to the development of new attitude estimation schemes that utilize the global representation of attitude using rotation matrices, free from the singularities associated with coordinates. In the 1960s, Schweppe proposed estimation schemes based on deterministic bounding of measurement and process uncertainty[4]. Maksarov and Norton continued this research by proposing to use ellipsoidal bounds to characterize the measurement and process noise [5]. The benefits of such deterministic estimation schemes are that it can use any representation for the estimate and requires no information about the noise distribution within the bounding ellipsoid. Related work has dealt with to proper identification of parameters in the noise model [6].

In the field of spacecraft this scheme has been applied to collision avoidance between satellites[7]. Applying this deterministic estimation scheme to spacecraft attitude estimation was proposed and verified by Sanyal et al. [8]. This estimation scheme will be used on Hawai‘iSat-1 which is being built by the Hawai‘i Spaceflight Laboratory at the University of Hawai‘i at Mānoa for launch in late 2012. This small satellite requires pointing and tracking for imaging and communication. The satellite contains three imaging payloads, two digital cameras for acquiring images of separation, the earth and the moon; and one thermal hyperspectral imager which acquires an “image” that can be used to determine the chemical makeup of targets. The thermal hyperspectral imager acquires an image by sweeping over a the length of the target over a matter of seconds. For this maneuver, the attitude of the spacecraft must be controlled. The satellite also contains two C-band radar transponders which require a pointing accuracy of three degrees.

To reduce some risk in implementing a new estimation scheme with no flight heritage, the work described in this document was completed to compare the performance of this
deterministic estimation scheme with that of an Extended Kalman Filter. The system chosen for this comparison was the spherical pendulum which is analogous to a satellite with attitude control about two body axes for pointing control. A satellite model is acted upon by a changing gravity field and other disturbances. The spherical pendulum acts under a constant gravity vector and has no other forces acting upon it. In this comparison the two dimensional representation of the reduced attitude and the two dimensional angular velocities are represented by spherical coordinates in the case of the EKF and by a reduced attitude vector and the angular velocity vector (global representation) for the deterministic estimation scheme. The results from the comparison should allow the Hawai’iSat-1 team to respond to criticism with evidence of comparable or better performance of the deterministic estimation scheme compared with the popular EKF.

This document is arranged as follows: Chapter 2 provides a detailed description of the spherical pendulum system and defines nomenclature that will be used throughout the document. The equations of motion for both the global and spherical coordinate representations are derived in Chapter 2. Chapter 3 describes the theoretical basis for both the deterministic estimation scheme and the Extended Kalman Filter. In this same chapter the specific implementations for these schemes for the spherical pendulum system are described. Chapter 3 also includes the method by which the two different estimation schemes are compared and, specifically, how a similar noise model is applied to both for a fair comparison. Chapter 4 details the development process for the simulation code and the verification steps that are taken to ensure proper results for the comparison. Chapter 5 presents detailed results of the comparison and a discussion of those results. Finally, Chapter 6 summarizes the work completed and proposes future work that may be carried out for the comparison of the the two estimation schemes.
In this chapter the physical description of the spherical pendulum system is given. The dynamics models for the spherical pendulum used for both the Extended Kalman Filter and the deterministic estimation scheme are presented along with their derivations. Other representations are discussed. The equations of motion are presented for each of the representations. The physical description of the pendulum is presented.

For this comparison, the two estimation schemes were applied to the attitude and angular velocities of a spherical pendulum. A one dimensional planar pendulum contains a mass particle that is constrained to move only within a plane and therefore has one variable dimension. A two dimensional spherical pendulum (such as the one used in this comparison) is constrained to move along the surface of a sphere, $S^2$, embedded in three dimensional space, $\mathbb{R}^3$; this is a two degree of freedom problem. The three dimensional pendulum has three degrees of rotational freedom, the two present for the spherical pendulum and then also rotation about its own body axis. The three dimensional pendulum is described in the configuration space $SO(3)$[9].

### 2.1 Representations

There are many ways to describe the configuration, which will be referred to as attitude, of the spherical pendulum but each has its drawbacks.

For the 3D pendulum, where rotation about the body axis is part of the state definition, the attitude is described using a rotation matrix, $R^{3 \times 3}$ or just $R$, which describes the transformation from the body frame of the pendulum to the inertial frame, as shown in Figure 2.1, and whose transpose (which is also the inverse) describes the transformation from the inertial (blue coordinate frame) to the body frame (red coordinate frame). A rotation matrix has three independent components, but is represented by 9 parameters with 6 constraints. [10].
As previously stated, the spherical pendulum is a two degree of freedom system, therefore the entire rotation matrix is not necessary. It is sufficient to describe one vector of the inertial frame in the body frame and it makes sense for that vector to be the gravity vector, here described as the Z-axis of the inertial frame. Herein this is described as the reduced attitude vector, $\Gamma$ and is defined as:

$$\Gamma = R^T e_3$$

where $e_3 = [0 \ 0 \ 1]^T$ is the third unit vector of the inertial frame which is also parallel to the direction of uniform gravity.

The definition of $\Gamma$ in terms of spherical coordinates can be found by a composite of the elementary Euler angle rotations, this is a 3-2-1 Euler angle rotation from the body frame
to the inertial frame.

\[ \Gamma = R^T e_3 \]  
\[ \Gamma = \left( R_3 (\gamma) R_2 (\theta) R_1 (\phi) \right)^T e_3 \]
\[ \Gamma = R_1^T (\phi) R_2^T (\theta) R_3^T (\gamma) e_3 \]
\[ \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & \sin \theta & \cos \gamma & -\sin \gamma & 0 \\ 0 & 1 & 0 & \sin \gamma & \cos \gamma \end{bmatrix} \begin{bmatrix} 0 \\ \sin \phi & \cos \phi & 0 & 0 & 1 \end{bmatrix} \]
\[ \begin{bmatrix} \cos \gamma \cos \theta \\ \sin \phi \cos \gamma \sin \theta + \cos \phi \sin \gamma \\ \sin \phi \sin \gamma - \sin \theta \cos \phi \cos \gamma \end{bmatrix} \begin{bmatrix} -\sin \gamma \cos \theta \\ \cos \phi \cos \gamma \sin \theta + \sin \phi \cos \gamma \end{bmatrix} \begin{bmatrix} 0 \\ \sin \phi \cos \theta \\ \cos \phi \cos \theta \end{bmatrix} \]

Note that \( \gamma \) does not appear in the definition of the reduced attitude vector. The rotations \( \theta, \phi, \) and \( \gamma \) occur as seen in Figure 2.2. In the coordinate system presented, \( \phi \) is a rotation about the X-axis, \( \theta \) is a rotation about the Y-axis, and \( \gamma \) is the rotation about the Z-axis, which is not part of the state in a spherical pendulum system.

The reduced attitude describes part of the state of motion. The other component is given by the angular velocity. For the global (full attitude) representation, angular velocity is represented by \( \omega \) which is a vector consisting of the angular velocities about each of the
Figure 2.2: Spherical Coordinates

body frame axis:

\[
\omega = \begin{bmatrix}
\omega_1 \\
\omega_2 \\
\omega_3
\end{bmatrix}
\]

However, for the spherical pendulum case the third component of angular velocity is inconsequential and will remain constant (will be shown in section 2.2). For the sake of simplicity, the constant value is chosen to be 0. In spherical coordinates, angular velocity is defined by only \(\omega_1\) and \(\omega_2\). The complete representation of the state of the spherical pendulum system is \(x = \begin{bmatrix} \Gamma \omega \end{bmatrix}^T\) for the global representation and \(x = \begin{bmatrix} \theta \phi \omega_1 \omega_2 \end{bmatrix}^T\).

2.2 Physical Description

The physical characteristics of the pendulum are represented by its mass, \(m\), and the inertia matrix, \(J \in \mathbb{R}^{3 \times 3}\). The location of the center of mass described in the body frame is, \(\rho = [0 \ 0 \ \rho_s] \text{ m}\). The acceleration of gravity is represented by \(g\). For the simulation that is conducted for this comparison \(m = 1\text{kg}, \ g = 9.8\frac{m}{s^2}\) and \(J = \text{diag}[2\ 2\ 3]\).
2.3 Derivation of Continuous Equations of Motion

2.3.1 Global Representation

As previously stated, the angular velocity vector, $\omega$, is the vector whose components are the angular velocities of the three principle axes of body frame [10]:

$$\dot{R} = R\omega^\times$$  \hspace{1cm} (2.3)$$

where

$$\omega^\times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ -\omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$$

and $(\omega^\times)^T = -\omega^\times$

Using these relationships, $\dot{\Gamma}$ in terms of $\omega$ and $\Gamma$ is found by:

$$\dot{\Gamma} = \dot{R}^T e_3$$  \hspace{1cm} (2.4)$$

$$= (R\omega^\times)^T e_3$$

$$= -\omega^\times R^T e_3$$

$$= -\omega^\times \Gamma$$

Therefore,

$$\dot{\Gamma} = \Gamma \times \omega$$  \hspace{1cm} (2.5)$$

The dynamics equation can be found by using Euler’s equation for rigid body dynamics where the time derivative of the inertial angular momentum ($L = J\omega$) is equal to the applied moment, which in this case is, the moment due to gravity expressed in the inertial
coordinate frame \((M_g = mg \rho \times \Gamma)\).

\[
\frac{d}{dt} (RL) = \frac{d}{dt} (R (J \omega)) = RM_g
= R \left( \frac{d}{dt} ((J \omega)) + \omega \times J \omega \right)
= R (J \dot{\omega} - J \omega \times \omega) = Rmg \rho \times \Gamma
\]

The dynamics equation becomes [8]:

\[
J \dot{\omega} = J \omega \times \omega + m g \rho \times \Gamma \tag{2.7}
\]

Theorem 1: The angular momentum components along the symmetry axis and along the instantaneous direction of gravity denoted by \(\Gamma\), are both conserved for the spherical pendulum system given by (2.5) and (2.7). Using \(J = \text{diag}[J_t \quad J_t \quad J_a]\), the axially symmetric model gives rise to the "spherical" pendulum instead of the three-dimensional pendulum (2.7) becomes

\[
\begin{bmatrix}
J_t \dot{\omega}_1 \\
J_t \dot{\omega}_2 \\
J_a \dot{\omega}_3
\end{bmatrix} = \begin{bmatrix}
(I_t - J_a) \omega_2 \omega_3 - m g \rho \Gamma_2 \\
(I_a - J_t) \omega_3 \omega_1 + m g \rho \Gamma_1 \\
0
\end{bmatrix} \tag{2.8}
\]

### 2.3.2 Spherical Coordinates

For the Extended Kalman Filter propagation, the equations of motion need to be of the form:

\[
\begin{bmatrix}
\dot{\theta} \\
\dot{\phi} \\
\dot{\omega}_1 \\
\dot{\omega}_2
\end{bmatrix} = f \left( \theta \phi \omega_1 \omega_2 \right) \tag{2.9}
\]

By substituting the spherical coordinates into \(R\) and taking the time derivative, equation
(2.3) provides an expression for $\dot{\theta}$, $\dot{\phi}$, $\dot{\gamma}$ in terms of $\theta$, $\phi$, and $\omega$ components.

$$\omega^x = R^T \dot{R}$$

becomes

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} 0 & -1 & \sin \theta \\ -\cos \phi & 0 & \sin \phi \cos \theta \\ \sin \phi & 0 & -\cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\phi} \\ \dot{\gamma} \end{bmatrix}$$ (2.10)

The first two equations of motion are found by inverting this matrix:

$$\begin{bmatrix} \dot{\theta} \\ \dot{\phi} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} 0 & -\cos \phi & -\sin \theta \\ -1 & -\sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \sin \phi \sec \theta & -\cos \phi \sec \theta \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$$ (2.11)

Note that $\gamma$ does not contribute to either $\dot{\theta}$ or $\dot{\phi}$.

For the next two equations of motion, it is only necessary to substitute (2.2) in to (2.8) and solve for the components of $\dot{\omega}$:

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} = \begin{bmatrix} \frac{(J_r - J_a)\omega_2 \omega_3}{J_t} + \frac{mg \rho_s \sin \phi \cos \theta}{J_t} \\ \frac{(J_a - J_t)\omega_1 \omega_3}{J_t} + \frac{mg \rho_s \sin \theta}{J_t} \\ 0 \end{bmatrix}$$ (2.12)

Note that $\gamma$ is not present in these equations either.

Recall that for simplicity $\omega_3 = 0$, then the equations of motion can be re-written as:
\[
\begin{align*}
\dot{\theta} &= -\omega_2 \cos \phi \\
\dot{\phi} &= -\omega_1 - \omega_2 \sin \phi \tan \theta \\
\dot{\omega}_1 &= \frac{mg \rho_s \sin \phi \cos \theta}{J_t} \\
\dot{\omega}_2 &= \frac{mg \rho_s \sin \theta}{J_t}
\end{align*}
\tag{2.13}
\]

These are the continuous time equations of motion for the extended Kalman Filter. These equations do not contain singularities that would be present if the state representation were in terms of \( \dot{\theta} \) and \( \dot{\phi} \). In equation (2.13), \( \dot{\phi} \) contains \( \tan \theta \) which is actually a function of \( \cos \theta \) in the denominator which causes the singularity.
CHAPTER 3
ESTIMATION SCHEMES

In this chapter, a detailed description is given of the two estimation schemes used in this comparison study. Theory behind both the deterministic estimation scheme and the Extended Kalman Filter (EKF) is discussed and derivations of the critical equations are presented. The chapter concludes with a discussion of the methods for comparing the two disparate estimators. State estimation schemes all are designed to provide an estimate of the state and some information about the uncertainty of the estimate with time. This is done by starting with an initial estimate with some initial uncertainty, using a model to propagate the estimate and the uncertainty, then using a state measurement to update the estimate at the time a measurement is available. The method by which the propagation and update steps are carried out, along with the representation of uncertainty, is what characterizes an estimator.

3.1 Deterministic Estimation Scheme

The deterministic estimation scheme uses the global representation for the state estimate. This estimation scheme also uses an ellipsoid described by a matrix to express the uncertainty associated with the estimate. The matrix (and thereby the ellipsoid) is propagated along with the state estimate for each time step. Sensor measurements of the state also have an associated uncertainty ellipsoid which is characterized by the sensor and is not dependent on the reading itself. At the time instant when a sensor measurement is available the propagated estimate and ellipsoid are updated by taking the intersection of the propagated ellipsoid centered about the propagated state estimate, with the sensor ellipsoid centered about the sensor measurement. The smallest ellipsoid that encompasses the intersection volume is the new uncertainty ellipsoid and the center of that ellipsoid is the new state estimate.
3.1.1 Uncertainty Ellipsoids

For the spherical pendulum system described in Chapter 2 the state is represented by \( x \in \mathbb{R}^6 \), a six dimensional vector. The corresponding uncertainty matrix is represented by \( P \in \mathbb{R}^{6\times6} \), a symmetric, positive definite, six by six matrix. The general equation of the ellipsoid in terms of the state estimate and the uncertainty matrix is:

\[
E_{\mathbb{R}^6}(x, P) = \{ y \in \mathbb{R}^6 | (y - x)^T P^{-1} (y - x) \leq 1 \} \quad (3.1)
\]

The uncertainly ellipsoid defines the error bounds placed on the state estimate; in other words, it is assumed that the unknown true state falls within the uncertainty ellipsoid described by \( P \) centered about \( x \).

For this state definition for the spherical pendulum, \( x = [\Gamma \omega]^T \), the ellipsoid, \( (y - x) \), has no validity for \( \Gamma \) because the error in \( \Gamma \) \((\delta \Gamma)\) should be such that \( \Gamma + \delta \Gamma \) also lies on the sphere \( S^2 \). The error describing the difference between the center of the ellipsoid, \( \Gamma_0 \), and a reduced attitude vector within the bounds of the ellipsoid, \( \Gamma \) must be defined in a different way, using \( \eta \) where:

\[
\Gamma = \exp (\eta^\times) \Gamma_0 \quad (3.2)
\]

In this definition \( \exp (\eta^\times) \) is a rotation matrix that transforms \( \Gamma_0 \) to \( \Gamma \). The error, \( \eta \in \mathbb{R}^3 \), contains two pieces of information: First, the magnitude of \( \eta (||\eta||) \) is the angle by which \( \Gamma_0 \) is rotated, and second, the unit vector of \( \eta (\hat{\eta} = \frac{\eta}{||\eta||}) \) is the axis about which the rotation takes place. For simplicity, the error in \( \omega \) will be defined \( \delta \omega = \omega - \omega_0 \).

The definition of the error ellipsoid in the form of (3.1) for the spherical pendulum is:

\[
E(\Gamma_0, \omega_0, P) = \{ \Gamma \in S^2, \omega \in \mathbb{R}^3 | \begin{bmatrix} \eta \\ \delta \omega \end{bmatrix} \in E_{\mathbb{R}^6}(0, P) \} \quad (3.3)
\]
3.1.2 Propagation of State Estimates Using a Variational Integrator

The discrete equations of motion that are used to propagate the state estimate between successive measurements are found in the same manner as the full attitude equations presented in [11] using methodology developed in [12].

The rotation $F_k$ is approximated by:

$$F_k \approx \exp \left( h \omega_k \times \right) \approx I + h \omega_k$$

(3.4)

Assuming a constant angular velocity over each time step, the full attitude evolves by:

$$R_{k+1} = R_k F_k$$

Using the relationship between $\Gamma$ and $R$, the discrete propagation equation for the reduced attitude (2.5) is found to be:

$$\Gamma_{k+1} = R_{k+1}^T e_3$$

$$= F_{k+1}^T R_k^T e_3$$

(3.5)

$$\Gamma_{k+1} = F_{k+1}^T \Gamma_k$$

(3.6)

where $h = t_{k+1} - t_k$ is the size of the time step and $I$ is the identity matrix and $\exp \left( h \omega_k \times \right)$ is the matrix exponential of $h \omega_k \times$ such that:

$$\exp \left( h \omega_k \times \right) = \sum_{k=1}^{\infty} \frac{1}{k!} \left( h \omega_k \times \right)^k$$

This rotation ensures that $R$ evolves on SO(3) and therefore $\Gamma$ evolves also on $S^2$, the unit sphere.
The modified inertia matrix is defined as $J_d = \frac{1}{2} \text{tr} [J I_3 - J]$ which is the measurement of the bodies rotational inertia. The following identity of $\omega$ is used:

$$(J \omega)^\times = \omega^\times J_d + J_d \omega^\times$$  \hspace{1cm} (3.7)

A defining relationship between $F_k$ and $\omega_k$ is imposed by inserting the approximation (3.4) into the first identity:

$$(J \omega_k)^\times = \omega_k^\times J_d - J_d (\omega_k^\times)^T$$

$$\approx \frac{1}{h} \left( (F_k - I) J_d - J_d (F_k^T - I) \right)$$

$$= \frac{1}{h} \left( F_k J_d - J_d F_k^T \right)$$

This can be rewritten as:

$$F_k J_d - J_d F_k^T = h (J \omega_k)^\times$$  \hspace{1cm} (3.9)

$F_k$ is found by solving (3.9) implicitly.

The complete discretized dynamics equation defined in [11] is:

$$J \omega_{k+1} = F_k^T J \omega_k + h M_{k+1} \times v_{k+1} + h M_{k+1} + h \tau_k$$  \hspace{1cm} (3.10)

where $v_{k+1}$ is the linear velocity of the center of mass which for this system is 0 and $M_{k+1}$ is the moment due to gravity, $M_{k+1} = m g \rho \times \Gamma_{k+1}$ and $\tau_k$ is the applied torques, also equal to 0 for this system.

The dynamics equation for this spherical pendulum system is:

$$J \omega_{k+1} = F_k^T J \omega_k + h m g \rho \times \Gamma_{k+1}$$  \hspace{1cm} (3.11)

These equations will be referred to as the propagation equations for the state estimate or the center of the ellipsoid, from time $t_k$ to time $t_{k+1}$. 

15
3.1.3 Propagation of Uncertainty Bounds on State Estimates

The previous section described the method by which the state (attitude and angular velocities) of the pendulum are propagated forward in time. However, the purpose of an estimator is not only to provide a state value for a time interval but also to provide some knowledge about the uncertainty of this estimate. For the deterministic estimator, the uncertainty is bound by an ellipsoid that is centered about the current state estimate. The size of the ellipsoid is a measurement of the uncertainty of the estimate. This ellipsoid, which is represented by the uncertainty matrix $P$ as given in equation (3.3), must also be propagated between measurements and this is done by the discrete linear flow matrix, $A$ [13]:

$$P_{k+1} = A_k P_k A_k^T$$  (3.12)

The flow propagation matrix is determined by the transition matrix to relate the variations in the state at time $k$ to those at time $k + 1$ linearly:

$$
\begin{bmatrix}
\eta_{k+1} \\
\delta \omega_{k+1}
\end{bmatrix} =
A_k
\begin{bmatrix}
\eta_k \\
\delta \omega_k
\end{bmatrix}
$$  (3.13)

$A_k$ linearizes the discrete nonlinear equations about the current state estimate.

In [14] the variational mapping for the full attitude representation is described. Below is a summary.

For the full attitude representation, the rotation matrix, $R$, from (3.5) is related to the propagator, $F$, by:

$$F_k = R_k^T R_{k+1}$$

The variation of the rotation matrix is given by:

$$\delta R_k = R_k \Sigma_k^X$$

where $\Sigma_k \in \mathbb{R}^3$. 

16
So the variation of $F$ is:

$$\delta F_k = F_k \Sigma_{k+1}^\times - \Sigma_k^\times F_k$$

(3.14)

Taking the variation of the discrete kinematics, (3.9) becomes:

$$h (J\delta \omega)^\times = \delta F_d - J_d \delta F_k^T$$

(3.15)

and then substituting (3.14) for $\delta F_k$ and simplifying, (3.15) becomes:

$$h (J\delta \omega)^\times = \{\text{tr } (F_k J_d) I_3 - F_k J_d\} F_k \Sigma_{k+1} + (h [F_k^T (J\omega_k)^\times + h M_{k+1} F_k^T W_k^T + F_k^T (J F_k^T)] \delta \omega_k$$

(3.16)

Re-arranging (3.16), $\Sigma_{k+1}$ can be obtained as:

$$\Sigma_{k+1} = h F_k^T (\text{tr } (F_k J_d) I_3 - F_k J_d)^{-1} F_k^T \Sigma_k$$

(3.17)

Substituting equation (3.17) into equation (3.14), $\delta F_k$ can be described in terms for one time step $k$:

$$\delta F_k = h \left( \text{tr } (F_k J_d) I_3 - F_k J_d \right)^{-1} J \delta \omega_k$$

(3.18)

Then taking first variations on both sides of the discrete dynamics equation, (3.11), and substituting equation (3.18) for $\delta F_k$ gives:

$$J \delta \omega_{k+1} = h M_{k+1} F_k^T \Sigma_k$$

$$+ \left( h [F_k^T (J\omega_{k})^\times + h M_{k+1} F_k^T W_k^T + F_k^T (J F_k^T)] \delta \omega_k$$

(3.19)

where $W_k = \text{tr } (F_k J_d) I_3 - F_k J_d)^{-1} I$

and $M_{k+1} = mg \rho \Sigma_{k+1}^\times$

Combining equation (3.19) and equation (3.17) linearized discrete equations of motion
can be represented as:

\[
\begin{bmatrix}
\Sigma_{k+1} \\
\delta\omega_{k+1}
\end{bmatrix} = \begin{bmatrix}
\mathcal{A}_k & \mathcal{B}_k \\
\mathcal{C}_k & \mathcal{D}_k
\end{bmatrix} \begin{bmatrix}
\Sigma_k \\
\delta\omega_k
\end{bmatrix}
\]  

(3.20)

where:

\[
\mathcal{A}_k = F_k^T
\]

\[
\mathcal{B}_k = hF_k^T \{ \text{tr} (F_k J_d I_3 - F_k J_d)^{-1} J \}
\]

\[
\mathcal{C}_k = h J^{-1} \mathcal{M}_{k+1} F_k^T
\]

\[
\mathcal{D}_k = J^{-1} \left( h \left[ F_k^T (J\omega_k)^\times + h M_{k+1} F_k^T \right] W_k + F_k^T \right)
\]

\[
\mathcal{W}_k = \{ \text{tr} (F_k J_d I_3 - F_k J_d)^{-1} J \}
\]

and

\[
\mathcal{M}_{k+1} = mg\rho^x \Gamma_{x_{k+1}}^\times
\]

The matrix made up of \( \mathcal{A}_k, \mathcal{B}_k, \mathcal{C}_k, \) and \( \mathcal{D}_k \) in equation (3.20) is used to linearly propagate the uncertainty ellipsoid for the full attitude dynamics. The propagation matrix for the reduced attitude case, \( A_k \), is found by first relating \( \delta\Gamma \) to \( \Sigma \):

\[
\delta\Gamma_k = \delta R_k^T e_3 = \left( R_k \Sigma_k^x \right)^T e_3
\]

\[
= -\Sigma_k^x R_k^T e_3 = -\Sigma_k^x \Gamma_k
\]

\[
= \Gamma_k \times \Sigma_k
\]  

(3.21)

Substituting this relationship into (3.20) the relationship becomes:

\[
\begin{bmatrix}
\delta\Gamma_{k+1} \\
\delta\omega_{k+1}
\end{bmatrix} = \begin{bmatrix}
\Gamma_{k+1}^x & 0 \\
0 & I
\end{bmatrix} \begin{bmatrix}
\mathcal{A}_k & \mathcal{B}_k \\
\mathcal{C}_k & \mathcal{D}_k
\end{bmatrix} \begin{bmatrix}
\Sigma_k \\
\delta\omega_k
\end{bmatrix}
\]  

(3.22)

From equation (3.22), \( \delta\Gamma_{k+1} \) and \( \delta\omega_{k+1} \) in terms of \( \Sigma_k \) and \( \delta\omega_k \) is:

\[
\delta\Gamma_{k+1} = \Gamma_{k+1}^x F_k^T \Sigma_k + h \Gamma_{k+1}^x F_k^T W_k \delta\omega_k
\]  

(3.23)

and
\[
\delta \omega_{k+1} = hJ^{-1} M_{k+1} \Gamma_k^\times F_k^T \Sigma_k + D_k \delta \omega_k
\]  \hspace{1cm} (3.24)

Substituting the definition of \( M_{k+1} \) into (3.24), \( \delta \omega_{k+1} \) is now:

\[
\delta \omega_{k+1} = hJ^{-1} mg \rho^\times \Gamma_k^\times F_k^T \Sigma_k + D_k \delta \omega_k
\]  \hspace{1cm} (3.25)

In equations (3.23) and (3.25) terms involving \( \Sigma_k \) can be expressed in terms of \( \delta \Gamma_k \) as follows:

\[
\Gamma_k^\times F_k^T \Sigma_k = R_k^T e_3 R_k e_3 R_k \delta \omega_k
\]  \hspace{1cm} (3.26)

Substituting equation (3.26) into equations (3.23) and (3.25), the complete linear mapping for the reduced attitude dynamics is found to be:

\[
\begin{bmatrix}
    \delta \Gamma_{k+1} \\
    \delta \omega_{k+1}
\end{bmatrix} =
\begin{bmatrix}
    F_k^T \\
    hJ^{-1} mg \rho^\times F_k^T
\end{bmatrix}
\begin{bmatrix}
    \Gamma_k^\times \\
    h \Gamma_k^\times W_k
\end{bmatrix}
\begin{bmatrix}
    \delta \Gamma_k \\
    \delta \omega_k
\end{bmatrix}
\]  \hspace{1cm} (3.27)

where \( \eta \) is the small rotation corresponding to \( \delta \Gamma_k \), such that:

\[
\delta \Gamma_k = -\Gamma_k^\times \eta_k
\]  \hspace{1cm} (3.28)

This results in the expression for \( \eta \): \( \eta = \Gamma_k \times \delta \Gamma_k \), so:

\[
\begin{bmatrix}
    \eta_{k+1} \\
    \delta \omega_{k+1}
\end{bmatrix} =
\begin{bmatrix}
    \Gamma_k^\times \\
    0
\end{bmatrix}
\begin{bmatrix}
    F_k^T \\
    h \Gamma_k^\times W_k
\end{bmatrix}
\begin{bmatrix}
    \Gamma_k^\times \\
    -\Gamma_k^\times W_k
\end{bmatrix}
\begin{bmatrix}
    \delta \Gamma_k \\
    \delta \omega_k
\end{bmatrix}
\]  \hspace{1cm} (3.29)

using \( \Gamma_k^\times = (F_k^T \Gamma_k)^\times = F_k^T \Gamma_k^\times F_k \) and \( \Gamma \times \Gamma \eta = -\eta \) if \( \eta \) is orthogonal to \( \Gamma \) which it is for the
shortest $\eta$:

$$
\begin{bmatrix}
\eta_{k+1} \\
\delta\omega_{k+1}
\end{bmatrix} =
\begin{bmatrix}
F_k^T & hF_k^T \left( \Gamma_{k+1}^T \right)^2 \mathcal{W}_k \\
-hJ^{-1}mg\rho \times F_k^T \Gamma_k^T & D_k
\end{bmatrix}
\begin{bmatrix}
\eta_k \\
\delta\omega_k
\end{bmatrix}
$$

(3.30)

from which $A_k$ in equation (3.13) is:

$$
A_k =
\begin{bmatrix}
F_k^T & hF_k^T \left( \Gamma_{k+1}^T \right)^2 \mathcal{W}_k \\
-hJ^{-1}mg\rho \times F_k^T \Gamma_k^T & D_k
\end{bmatrix}
$$

(3.31)

### 3.1.4 Filtering Update

The propagated estimate has an associated uncertainty ellipsoid as does the sensor measurement. When a sensor measurement is available, an updated estimate and uncertainty ellipsoid will be found from the intersection of the flow-propagated ellipsoid (from time $t_0$ to $t_N = t_0 + Nh$) and the measurement-based ellipsoid (at time $t_N = t_0 + Nh$) [13]. The process for the ellipsoid intersection for a two dimensional system is shown in figure 3.1. Logically if, as is assumed, the unknown true state falls within the ellipsoids described by the propagated ellipsoid and the sensor ellipsoid then it must fall within the volume of intersection which can be encompassed by a new ellipsoid that will (unless one of the propagated or sensor ellipsoids completely contains the other) be smaller than either of propagated or sensor ellipsoids, and thereby less uncertain.

Let $\mathcal{E}_{R^e}(x_1, P_1)$ and $\mathcal{E}_{R^e}(x_2, P_2)$ be two ellipsoids with nonempty intersection. There exist an ellipsoid $\mathcal{E}_{R^e}(\hat{x}, \hat{P})$ satisfying:

$$
\min \text{tr}(\hat{P}), \text{ such that } \mathcal{E}_{R^e}(x_1, P_1) \cap \mathcal{E}_{R^e}(x_2, P_2) \subset \mathcal{E}_{R^e}(\hat{x}, \hat{P}),
$$

(3.32)

Since the flow-propagated uncertainty ellipsoid is non-degenerate the generalization of (3.32) that we need is:

$$
\min \text{tr}(\hat{P}), \text{ such that } \mathcal{E}_{R^e}(x_1, P_1) \cap \mathcal{E}_{R^e}(x_2, P_2) \subset \mathcal{E}_{R^e}(\hat{x}, \hat{P}), \min \text{tr}(\hat{P})
$$

(3.33)
where $M = P^{-1}$

Formulated in the language of uncertainty ellipsoids on the surface of the sphere and using the superscripts $f$ and $m$ for the flow and measured states the new state estimate and uncertainty matrix $(\hat{\Gamma}, \hat{\omega}, \hat{P})$ are found by:

$$\min \text{tr}(\hat{P}), \text{ such that } \mathcal{E}(\Gamma_f, \omega_f, P_f) \cap \mathcal{E}(\Gamma_m, \omega_m, P_m) \subset \mathcal{E}(\hat{\Gamma}, \hat{\omega}, \hat{P})$$

The filtered states $(\hat{\Gamma}, \hat{\omega})$ form the updated state estimates at time $t_N$. The problem of intersecting uncertainty ellipsoids on the surface of the sphere must be converted to an equivalent problem of intersecting uncertainty ellipsoids on $\mathbb{R}^6$ in order to use the described approach.

For the intersection calculation, the center of the measurement ellipsoid is chosen as origin of $\mathbb{R}^6$, i.e.,

$$\mathcal{E}_{\mathbb{R}^6}(0, P_m).$$

The center of the flow propagated-based ellipsoid in $\mathbb{R}^6$, with origin in the center of the measurement based ellipsoid, is the “Euclidean” difference between the centers of the two
ellipsoids on the surface of the sphere. Therefore,

\[
x = \begin{bmatrix} \eta \\ \delta \omega \end{bmatrix},
\]

(3.34)

where

\[
\eta = \frac{\arccos(\Gamma^f \cdot \Gamma^m)}{\sin(\arccos(\Gamma^m \cdot \Gamma^f))} \Gamma^m \times \Gamma^f
\]

(3.35)

and

\[
\delta \omega = \omega^f - \omega^m
\]

(3.36)

However, this definition of \( \eta \) becomes unstable as \( \tilde{\Gamma} \cdot \Gamma^f \) approaches 1, so in that case it is necessary to determine \( \eta \) in terms of its Taylor Series Expansion [15]:

\[
\eta = \left(1 - \frac{\Gamma^m \cdot \Gamma^f}{3} - \frac{2(\Gamma^m \cdot \Gamma^f - 1)^2}{15} - \frac{2(\Gamma^m \cdot \Gamma^f - 1)^3}{35} + \frac{8(\Gamma^m \cdot \Gamma^f - 1)^4}{315}\right) \Gamma^m \times \Gamma^f
\]

Thus the intersection:

\[
\min \text{tr}(\hat{P}), \text{ such that } E_{\mathbb{R}^6}(\eta, \delta \omega, P^f) \cap E_{\mathbb{R}^6}(0, P^m) \subset E_{\mathbb{R}^6}(\hat{x}, \hat{P})
\]

(3.37)

Using the intersecting process laid out in [13] the filter-updated uncertainty ellipsoid in \( \mathbb{R}^6 \) is found to be the solution to (3.37).

The updated uncertainty ellipsoid in \( \text{SO}(3) \times \mathbb{R}^3 \) is obtained by translating this ellipsoid as \( E(\hat{\Gamma}, \hat{\omega}, \hat{P}) \) where the center in terms of \( \Gamma \) and \( \omega \) is obtained from \( \hat{x} = [\eta^T, \delta \omega^T]^T \in \mathbb{R}^6 \):

\[
\hat{\Gamma} = \exp(\hat{\eta}^\times) \Gamma^m,
\]

\[
\hat{\omega} = \tilde{\omega} + \delta \omega^m.
\]

(3.38)
3.2 Extended Kalman Filter

In this section the Extended Kalman Filter is described in detail in [16] which uses the spherical coordinate representation of the state. The Extended Kalman filter (EKF) differs from the Kalman Filter because it can be applied to nonlinear systems by linearizing about the point of interest at each time step. The EKF algorithm applied in this case is a Hybrid Extended Kalman Filter, so called because it uses a continuous time dynamics model and discrete time measurements.

3.2.1 Covariance Matrix

The uncertainty of the state estimation EKF is described by a covariance matrix $P$. This uncertainty matrix is initially made up of the covariances of each of the state variables:

$$P = \begin{bmatrix}
\sigma_\theta^2 & \sigma_\theta \sigma_\phi & \sigma_\theta \sigma_{\omega_1} & \sigma_\theta \sigma_{\omega_2} \\
\sigma_\theta \sigma_\phi & \sigma_\phi^2 & \sigma_\phi \sigma_{\omega_1} & \sigma_\phi \sigma_{\omega_2} \\
\sigma_\theta \sigma_{\omega_1} & \sigma_\phi \sigma_{\omega_1} & \sigma_{\omega_1}^2 & \sigma_{\omega_1} \sigma_{\omega_2} \\
\sigma_\theta \sigma_{\omega_2} & \sigma_\phi \sigma_{\omega_2} & \sigma_{\omega_1} \sigma_{\omega_2} & \sigma_{\omega_2}^2
\end{bmatrix}$$  \hspace{1cm} (3.39)

This assumes that all of the uncertainty in the state estimate is Gaussian and noise is normally distributed about the mean. This is, of course, an idealization. The covariance measurement is propagated along with state estimate. The sensor reading also has an associated covariance matrix that is characterized by the sensor.

3.2.2 Propagation

At its outset, the EKF propagates an initial state estimate forward in time to create an priori estimate for the state, $x_k^-$ at a moment in time, ‘a prior’ to a measurement being made. This propagation is accomplished via integration of the system equations and the ordinary
differential equation (ODE) describing the evolution of the covariance matrix \( P_k^- \):

\[
\dot{x} = f(x) \\
\dot{P} = AP + PA^T
\]

where \( f(x) \) are the continuous equations of motion, (2.13), and where \( A \) is the Jacobian for the system state equations, defined in [16]:

\[
A = \frac{\partial f}{\partial x}
\] (3.40)

\[
A = \begin{bmatrix}
0 & \omega_2 \sin \phi & 0 & -\cos \phi \\
-\omega_2 \sin \phi \sec^2 \theta & -\omega_2 \cos \phi \tan \theta & -1 & -\sin \phi \tan \theta \\
-gmp_c \sin \phi \sin \theta & gp_c \cos \phi \cos \theta & 0 & 0 \\
m gp_c \cos \theta & 0 & 0 & 0
\end{bmatrix}
\] (3.41)

### 3.2.3 Update

After taking a measurement, a posteriori estimate for the state of the system is generated by first finding the a priori covariance of the innovation \( S_k \) for this time step using the covariance matrix \( P_k^- \) and the covariance matrix of the sensor reading \( R_k \). The innovation is defined as the difference between the propagated estimate of the state and the sensor reading, and is designated \( z_k \). Next, the Kalman gain \( K_k \) is calculated, and finally the a posteriori estimate and covariance matrix \( x_k^+ \) and \( P_k^+ \) can be found:

\[
S_k = P_k^- + R_k \\
K_k = P_k^- S_k^{-1} \\
x_k^+ = x_k^- + K_k (z_k - x_k^-) \\
P_k^+ = (I + K_k -) P_k^-
\]
3.3 Methods For Comparison

For a fair comparison of the abilities of the two estimation schemes, two things must be true. First, both estimation schemes must have the same initial conditions, specifically the same initial state estimate and initial uncertainty for that estimate. Second, sensor readings and associated uncertainty must be the same.

Although the state representation for each of the estimators is different, it is not complicated to convert between them using (2.2). However creating a uncertainty model for one estimation scheme and converting it for the other is slightly more challenging.

As previously stated, the uncertainty model for the Extended Kalman Filter is optimal for a normally distributed error model where the state that is given as the estimate is the mean (and most likely true state). As states move away from the mean they are less and less likely to be the truth. The deterministic estimation scheme uses an ellipsoid where no distribution of uncertainty within it is provided (or needed). In other words, all the deterministic estimator is conveying is that the truth falls within a certain bound but does not specify the likelihood of the truth being in a particular part of that ellipsoid.

For this comparison, an ellipsoid whose radii are equal to three standard deviations (and thereby contains 99.6% of all possible values) of the noise is used. Using the method laid out in [17] the errors are $\delta \theta = 3\sigma_\theta$, $\delta \phi = 3\sigma_\phi$, $\delta \omega_1 = 3\sigma_{\omega_1}$, and $\delta \omega_2 = 3\sigma_{\omega_2}$.

Recall that the reduced attitude vector, $\Gamma = Re_3$, and the rotation matrix $R$ can be expressed as a composite rotation matrix of elementary rotation matrices such that it is a 2-1 Euler angle rotation:

$$R = R_1(\phi) R_2(\theta)$$

This is the rotation matrix that transforms $e_3$ to $\Gamma$. However, if instead, a reduced attitude vector located some ways away from this $\Gamma$ by $\delta \theta$ and $\delta \phi$ is desired, then the
following rotation matrix describes the rotation from $e_3$ to this desired vector $\Gamma'$:

$$R' = R_1 \left( \phi + \delta\phi \right) R_2 \left( \theta + \delta\theta \right)$$

The two reduced attitude vectors can be related by:

$$\Gamma' = R' e_3$$

$$= R' R^T \Gamma$$

The rotation matrix that transforms $\Gamma$ to $\Gamma'$ is $R'R^T$ and can be expressed as the matrix exponent of skew symmetric matrix of the vector $\eta$, so $\eta$ is found from the matrix logarithm of $R'R^T$:

$$\eta^{\times} = \log \left( R'R^T \right)$$

Using the errors as they are described above, this provides an $\eta$ vector for the error at a specific state ($\theta$ and $\phi$) and specified standard deviations $\sigma_\theta$ and $\sigma_\phi$. The first possible problem with this solution is that this $\eta$ is a rotation from $\Gamma$ to $\Gamma'$ but perhaps not shortest rotation. The shortest rotation is found by eliminating any part of the vector $\eta$ that is not perpendicular to $\Gamma$ and $\Gamma'$. In other words, the shortest possible rotation is one where $\Gamma$ and $\Gamma'$ are in a plane with normal vector $\eta$. This is found by:

$$\eta_{\text{short}} = \left( \eta^T \Xi \right) \Xi$$

where

$$\Xi = \frac{\Gamma \times \Gamma'}{\|\Gamma \times \Gamma'\|}$$

This can be directly calculated using the expression for $\Gamma$ in terms of $\theta$ and $\phi$. 

26
The expression for $\Gamma$ in terms of $\theta$ and $\phi$, (2.2) results in:

$$\delta \Gamma = \begin{bmatrix}
\delta \theta \cos \theta \\
-\delta \phi \cos \phi \cos \theta + \delta \theta \sin \phi \sin \theta \\
-\delta \phi \sin \phi \cos \theta - \delta \theta \cos \phi \sin \theta
\end{bmatrix}$$  \hspace{1cm} (3.43)

Using (3.28) and (3.43) and solving for the individual components of $\eta$:

$$\eta = \begin{bmatrix}
-\delta \phi \\
-\delta \theta \cos \phi \\
-\delta \theta \sin \phi
\end{bmatrix}$$  \hspace{1cm} (3.44)

However, it is still necessary to convert this $\eta$ into the shortest rotation:

$$\eta_{\text{short}} = \eta - (\eta^T \Gamma) \Gamma$$  \hspace{1cm} (3.45)

$$= \begin{bmatrix}
-\delta \phi \cos^2 \theta \\
-\delta \theta \cos \phi - \delta \phi \sin \theta \cos \theta \sin \phi \\
-\delta \theta \sin \phi + \delta \phi \sin \theta \cos \theta \cos \phi
\end{bmatrix}$$  \hspace{1cm} (3.46)

Another problem is that this particular $\eta$ is very dependent on the state ($\theta$ and $\phi$) it evaluated at, so although the standard deviations are constant, $\eta$ is changing because the state is changing. This is because these errors are two dimensional errors that are being represented in three dimensional space so at each state only two dimensions are necessary. Although the error for $\Gamma$ is actually an ellipsoid this method gives only an ellipse created by “slicing” the ellipsoid with a plane. To solve this problem $\eta$ will only be calculated in this manner for the initial condition. Then the largest of the three components will be used to define the initial uncertainty matrix for the estimate for the deterministic estimator:
\[
\begin{equation}
P_0 = \text{diag} \begin{bmatrix}
\eta_{\text{max}}^2 & \eta_{\text{max}}^2 & \delta \omega_{\text{max}}^2 & \delta \omega_{\text{max}}^2
\end{bmatrix}
\end{equation}
\] (3.47)

where \( \eta_{\text{max}} \) is the largest component of \( \eta_{\text{short}} \).

This uncertainty matrix has spherical regions for both \( \eta \) and \( \delta \omega \) but becomes non-spherical as it is propagated forward.

The same process is carried out to obtain the uncertainty matrix associated with the sensor measurements and uncertainty matrix is the same at each measurement.
CHAPTER 4
SIMULATION MODELS

This chapter focuses on the development of the simulation code used in the comparison and the tests that were applied to verify the code works properly.

4.1 Simulation Structure

This section describes the different parts of the code implementing the estimations.

4.1.1 Generating the “Truth”

In a real-world application of state estimation, the truth can never be known. However, for simulation purposes, a state is generated to act as the “truth”. The truth is generated by the global propagation equations from the initial state using (3.6), (3.9), and (3.11). The truth is propagated using a time step size that is 100 times smaller than the propagation time step size of the estimation schemes. The smaller time step yields a more accurate representation of the true state.

4.1.2 Initial Estimate and Sensor Reading Generation

The initial estimate is generated by first randomly generating an $\eta$ and $\delta \omega$ that exists inside the initial uncertainty ellipsoid defined by (3.47). These values can be selected using any distribution. Then the selected $\eta$ is converted to the shortest rotation using (3.46). Using the short $\eta$ and $\delta \omega$, the initial estimate is found by applying these errors to the true initial condition:

$$\Gamma_0 = \exp(\eta^\times) \Gamma_T$$

$$\omega_0 = \omega_T + \delta \omega$$

(4.1)
where $\Gamma_T$ and $\omega_T$ represent the true internal state.

Sensor readings are generated in much the same manner. However, $\eta$ and $\delta\omega$ are selected from within the sensor uncertainty ellipsoid and the error is applied to the truth at that time step.

### 4.1.3 Intersection

The goal of the intersection was stated in Chapter 3, however it is not simple to find the ellipsoid with the minimum trace that encompasses the volume of intersection. The code used for the finding the intersection ellipsoid is based on the solution to (3.33) given in [5].

The new uncertainty ellipsoid is $\hat{P}$ and the center of this ellipsoid, which corresponds to $[\hat{\eta}, \hat{\delta} \omega]$, is $\hat{x}$. The solution is:

$$\hat{x} = x_1 + \hat{L}(q_0) (x_2 - x_1)$$

$$\hat{P} = \beta(q_0) \left( I_n - \hat{L}(q_0) \right) P_1$$

where

$$\hat{L}(q) := P_1 \left( P_1 + q^{-1}P_2 \right)^{-1}$$

$$\beta(q) := 1 + q - (x_2 - x_1)^T P_1^{-1} \hat{L}(q) (x_2 - x_1)$$

and $P_1$ is always the smaller of the two uncertainty ellipsoids and $x_1$ is its center.

$q = q_0$ is the solution to:

$$\frac{\text{tr} \left( U^{-1} P_2 U \left( \Lambda + q I_n \right)^{-2} \right)}{\text{tr} \left( U^{-1} P_2 U \left( \Lambda + q I_n \right)^{-1} \right)} = \frac{\beta'(q)}{\beta(q)}$$

(4.4)
Where $U$ and $\Lambda$ such that $P_1^{-1}P_2 = U\Lambda U^{-1}$. Equation 4.4 can be re-written as:

$$\frac{\text{tr}(F(I - qP_2^{-1})P_2^{-1}P_1)}{\text{tr}((I - qFP - P_2^{-1})P_1)} = \frac{\beta'(q)}{\beta(q)}$$

In the same fashion the definition of $\beta$ can be re-written as:

$$\beta(q) = 1 + q - q(x_2 - x_1)^T P_1^{-1}FP_2(x_2 - x_1)$$  

(4.6)

where $F = (P_1^{-1} + qP_2^{-1})^{-1}$.

Then the derivative of $\beta$ with respect to $q$ is:

$$\beta'(q) = 1 - (x_2 - x_1)^T P_1^{-1}FP_2^{-1}FP_2^{-1} \left( I - qFP_2^{-1} \right) (x_2 - x_1)$$  

(4.7)

The value $q_0$ is found by iterating to a solution of (4.5) where the difference ($f$) between the left hand side and the right hand side is minimized. The initial value for $q$ is 0 and it is reduced by $f \left( \frac{df}{dq} \right)^{-1}$ until the absolute value of $f$ is less than $10^{-15}$.

**Special Cases**

Two special cases exist for intersecting. The first case is when one ellipsoid exists entirely inside the other. In this case the smaller of the two ellipsoids is the solution to (3.33). The smaller ellipsoids, $P_1$, and its center, $x_1$, are returned and no other calculations are carried out. The test for this case is:

$$\text{flag} = \left( 1 - (x_2 - x_1)^T P_2^{-1} (x_2 - x_1) \right) \text{tr}(P_1) - \text{tr}\left( P_1P_2^{-1}P_1 \right)$$  

(4.8)

If the flag is greater than 0, then $P_1$ is a subset of $P_2$.

The second special case is if there is no intersection at all. This will only occur if the truth no longer lies inside the uncertainty and usually is an indicator of an error in the ellipsoid definition or that the time between sensor readings is too large. This is the case if
the trace of the resulting uncertainty ellipsoid is negative. In the event that this does occur, the simulation chooses the sensor reading and uncertainty ellipsoids to be the new estimate and uncertainty ellipsoid and continues forward. This effectively starts the process over again with a new initial estimate and a smaller initial uncertainty.

### 4.2 Verification Tests

This section discusses some of the tests that were preformed to verify that the code was working correctly.

#### 4.2.1 Coordinate Conversion

The first test that is critical, but not particularly difficult, is the conversion test. This test ensures that the conversion from spherical coordinates to the global representation and the reverse conversion are correct. This test consists of:

1. Generating a series of values for the global representation by propagating an initial state

2. Converting those into the spherical coordinates using the convert2 function by:

\[
\begin{align*}
\theta &= \arcsin \Gamma_1 \\
\phi &= \arctan \frac{\Gamma_2}{\Gamma_3}
\end{align*}
\]

(4.9)

3. Converting those values back to the global representation using the convertF function based on equation (2.2)

4. Once again converting those values back to spherical coordinates

This results in two sets of values for each representation. Figure 4.1 shows the error for each parameter. All of the errors fall within the precision error of the computer.
4.2.2 Linearity Test of the Dynamics Model

The linearity test is performed to verify that the propagation code is correctly linearizing the model in (3.13).

The linearization is of the small variations ($\eta$ and $\delta \omega$) of the state variables ($\Gamma$ and $\omega$) such that:

$$\begin{bmatrix} y_{\Gamma_k} \\ y_{\omega_k} \end{bmatrix} = \begin{bmatrix} \Gamma_k + \epsilon \eta_k \\ \omega_k + \epsilon \delta \omega_k \end{bmatrix}$$

(4.10)

For the sake of simplicity, the propagation functions (3.6) and (3.11) do not need to be re-written in their entirety. They need only be expressed as:

$$\begin{bmatrix} \Gamma_{k+1} \\ \omega_{k+1} \end{bmatrix} = f \left( \begin{bmatrix} \Gamma_k \\ \omega_k \end{bmatrix} \right)$$

(4.11)

Using the relationships defined in (4.10) and (4.11), the propagation function can be re-written as:

$$\begin{bmatrix} \Gamma_{k+1} + \epsilon \eta_{k+1} \\ \omega_{k+1} + \epsilon \delta \omega_{k+1} \end{bmatrix} = f \left( \begin{bmatrix} \Gamma_k + \epsilon \eta_k \\ \omega_k + \epsilon \delta \omega_k \end{bmatrix} \right)$$

(4.12)
Using the Taylor Series Expansion to approximate the function, the relationship becomes:

$$
\begin{bmatrix}
\Gamma_{k+1} + \epsilon\eta_{k+1} \\
\omega_{k+1} + \epsilon\delta\omega_{k+1}
\end{bmatrix} = f\left(\begin{bmatrix}
\Gamma_k \\
\omega_k
\end{bmatrix}\right) + A_k\epsilon\begin{bmatrix}
\eta_k \\
\delta\omega_k
\end{bmatrix} + O\left(\epsilon^2\right)
$$

(4.13)

where $O$ represents the higher order terms.

Solving for these higher order terms the definition becomes:

$$
P(\epsilon) = f\left(\begin{bmatrix}
\Gamma_k + \epsilon\eta_k \\
\omega_k + \epsilon\delta\omega_k
\end{bmatrix}\right) - f\left(\begin{bmatrix}
\Gamma_k \\
\omega_k
\end{bmatrix}\right) + \epsilon A_k\begin{bmatrix}
\eta_k \\
\delta\omega_k
\end{bmatrix} = O\left(\epsilon^2\right)
$$

(4.14)

The test steps are as follows:

1. Defining the base state at time $t_0$ is $\Gamma_0$ and $\omega_0$
2. Defining a base error at time $t_0$, $\eta_0$ and $\delta\omega_0$
3. Defining two functions, $\Gamma_0(\epsilon) = \exp(\epsilon\eta_0)$ and $\omega_0(\epsilon) = \epsilon\delta\omega_0 + \omega_0$
4. Propagating these functions to time $t_1$ using (3.6), (3.9), and (3.11) such that

$$
[\Gamma_1(\epsilon), \omega_1(\epsilon)] = \text{Propagate}([\Gamma_0(\epsilon), \omega_0(\epsilon)])
$$

Note that the base state at $t_1$ is $\Gamma_1 = \Gamma_1(1)$ and $\omega_1 = \omega_1(1)$

5. Propagating the original $\eta_0$ and $\delta\omega_0$ using (3.13) but scaling the values with coefficient $\epsilon$ such that:

$$
\begin{bmatrix}
\eta_1(\epsilon) \\
\delta\omega_1(\epsilon)
\end{bmatrix} = A_k\epsilon\begin{bmatrix}
\eta_0 \\
\delta\omega_0
\end{bmatrix}
$$

6. Find errors, $\tilde{\eta}_1(\epsilon)$ and $\tilde{\delta}\omega_1(\epsilon)$ for each $\Gamma_1(\epsilon)$ and $\omega_1(\epsilon)$ with respect to the propagated base state $\Gamma_1$ and $\omega_1$ using (3.35) and (3.36)

7. Defining a function, $P(\epsilon) = \|\begin{bmatrix}
\tilde{\eta}_1(\epsilon) - \eta_1(\epsilon) \\
\tilde{\delta}\omega_1(\epsilon) - \delta\omega_1(\epsilon)
\end{bmatrix}\|$
Plotting $P(\epsilon)$ for values of $\epsilon$ ranging from 0.1 to 1 on a logarithmic scale should result in a line of slope $\log(2)$ if the linearization $A_k$ is correct because the higher order terms are of the second order as seen in (4.14). The results are shown in Figure 4.2 which shows that the linearization is correct.

![Figure 4.2: Linearization Test of $A_k$](image)

### 4.2.3 Chaos Test

The chaos test is a short test of the chaos of a system by measuring when a set of initial states that are originally bounded by a small ellipsoid move outside a larger ellipsoid. The test is carried out as follows:

1. Selecting a center state and an uncertainty ellipsoid
2. Generating ten states that are at the boundary of the uncertainty ellipsoid
3. Propagating the center state, the ten boundary states and an ellipsoid that is four times the size of the original ellipsoid
4. Note the time when the boundary states move outside the larger ellipsoid
The shorter the time before the states move outside the larger uncertainty ellipsoids, centered about the center state, the more chaotic the system is. Figure 4.3 shows the results for each of the boundary states at each time step beginning at the inverted position. When the test value becomes greater than 1, the state has moved outside the larger ellipsoid. The first of the boundary states to cross outside of the uncertainty ellipsoid is at about 0.7 seconds. This means that the deterministic estimation scheme may not work properly if the sensor readings are more than 0.7 seconds apart. After 0.7 seconds, it is possible that the truth will no longer be within the uncertainty ellipsoid centered about the estimate. It is also possible that the intersection will not a yield a positive volume because the propagated ellipsoids and the sensor ellipsoid do not intersect.
CHAPTER 5
RESULTS AND ANALYSIS

This chapter presents the results of the comparison simulation for Gaussian and Uniform noise distribution models.

5.1 Initial Conditions

Sensor readings are generated at an interval of 0.5 seconds and the estimate is propagated with 0.01 second time steps.

Two sets of initial conditions are used: one simulation starts with the initial state hanging seen in figure 5.1 and the other with the initial state at the inverted position seen in figure 5.2. The initial state values for both positions can be found in 5.1.

<table>
<thead>
<tr>
<th>State Component</th>
<th>Hanging</th>
<th>Inverted</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Gamma_0$</td>
<td>$[0\ 0\ 1]^T$</td>
<td>$[0\ 0\ -1]^T$</td>
</tr>
<tr>
<td>$\omega_0^2/\omega$</td>
<td>$[17.2\ -28.6\ 0]^T$</td>
<td>$[17.2\ -28.6\ 0]^T$</td>
</tr>
</tbody>
</table>

Table 5.1: Initial States

Figure 5.1: Pendulum in Hanging Position
For generating the initial uncertainty matrix the standard deviations are found in 5.2:

\[
\begin{array}{ccc}
\text{Standard Deviations} & \text{Initial Estimate} & \text{Sensor Measurement} \\
\sigma_\theta & 10^\circ & 2.5^\circ \\
\sigma_\phi & 10^\circ & 2.5^\circ \\
\sigma_{\omega_1} & 10^\circ/s & 2.5^\circ/s \\
\sigma_{\omega_2} & 10^\circ/s & 2.5^\circ/s \\
\end{array}
\]

Table 5.2: Noise Model Definition

5.2 Gaussian Noise Model Comparison

The first comparison is conducted using a Gaussian distribution for the noise within the uncertainty ellipsoid, as seen in figure 5.3 where the values on the x-axis are the standard deviations. The noise is generated using the randn function in Matlab which uses a Gaussian distribution to return a number between -1 and 1:

\[
v = \sqrt{\frac{1}{3}} P \cdot \text{randn}(1,6)
\]

where \(\sqrt{}\) takes the square root of the matrix and \(P\) is the uncertainty matrix for the
deterministic estimation scheme.

(Recall that three times the standard deviation was used to map the uncertainty in the spherical coordinates to the global representation uncertainty.)

Figures 5.4 - 5.7 show the state components for each estimation scheme for two simulations starting from the inverted position but with different (randomly selected initial estimates).

The actual measure of the quality of the estimate is not seen in figures 5.4 and 5.7 but rather in the measure of the error of both of the estimation schemes. Figures 5.8 and 5.10 compare the error for both the deterministic estimator and the Extended Kalman Filter with $||\delta \omega||$, the magnitude of the difference of the angular velocity vectors of the truth and the estimate, and $||\eta||$, the angle between the truth and the estimated reduced attitude vectors. While Figures 5.9 and 5.11 show the size of the uncertainty ellipsoid for the deterministic estimation scheme and nine times the size of the covariance matrix for the EKF so as to encompass 99.6% of the distribution. It can be seen that the deterministic estimator yields less error throughout the simulation for the second simulation but not for the first; the performance is very dependent of the how far away the initial estimate is from the initial
true state.

Figure 5.12 and 5.13 shows the state components for each estimation scheme starting from the hanging position. The motion is much more contained and the EKF performs quite well.
Figure 5.6: Global Representation Components for EKF for Inverted Initial State and Gaussian Noise (Second Simulation)

Figure 5.7: Global Representation Components for D.E. for Inverted Initial State and Gaussian Noise (Second Simulation)
Figure 5.8: Error for the Inverted Initial Condition and Gaussian Noise

Figure 5.9: Uncertainty for the Inverted Initial Condition and Gaussian Noise

Figure 5.14 compares the error for both the deterministic estimator and the Extended Kalman Filter and 5.15 shows the uncertainty ellipsoid size. The performances of the
Figure 5.10: Error for the Inverted Initial Condition and Gaussian Noise (Second Simulation)

Figure 5.11: Uncertainty for the Inverted Initial Condition and Gaussian Noise (Second Simulation)
Figure 5.12: Global Representation Components for EKF for Hanging Initial State and Gaussian Noise

Figure 5.13: Global Representation Components for D.E. for Hanging Initial State and Gaussian Noise
two estimation schemes are much more similar for the hanging condition. The EKF even
slightly outperforms the deterministic estimator for very localized motion.

Figure 5.14: Error for the Hanging Initial Condition and Gaussian Noise

Figure 5.15: Error for the Hanging Initial Condition and Gaussian Noise
5.3 Uniform Noise Model Comparison

The second comparison was completed using uniformly distributed noise within the uncertainty ellipsoids, as seen in figure 5.16 where the values on the x-axis are the standard deviations, recall that the ellipsoid bounds were set to three times the standard deviation, therefore the uniform distribution is from $-3\sigma$ to $3\sigma$. The uniform noise is created using the rand function in Matlab which returns a number between 0 and 1 using a uniform distribution:

$$ v = \sqrt{\frac{1}{3} P (2 \cdot \text{rand}(1, 6) - 1)} $$

Figure 5.16: Noise Distributions for simulation

Figures 5.17 - 5.20 show the state components for each estimation scheme for two simulations. The EKF appears to deal quite poorly with the uniformly distributed noise and the large range of motion.

Figures 5.21 and 5.23 show the errors for both the estimation schemes for the inverted initial condition and uniformly distributed error for both simulations and figures 5.22 and 5.24 show the uncertainty ellipsoid size.
Figure 5.17: Global Representation Components for EKF for Inverted Initial State for Uniformly Distributed Noise

Figure 5.18: Global Representation Components for D.E. for Inverted Initial State for Uniformly Distributed Noise
Figure 5.19: Global Representation Components for EKF for Inverted Initial State for Uniformly Distributed Noise (Second Simulation)

Figure 5.20: Global Representation Components for D.E. for Inverted Initial State for Uniformly Distributed Noise (Second Simulation)
Figures 5.25 and 5.26 show the state components for each estimation scheme.

Figure 5.27 shows the errors for both the estimation schemes for the hanging initial condition.
Figure 5.23: Error for the Inverted Initial Condition Using Uniformly Distributed Noise (Second Simulation)

Figure 5.24: Uncertainty for the Inverted Initial Condition Using Uniform Distributed Noise (Second Simulation)
Figure 5.25: Global Representation Components for EKF for Hanging Initial State for Uniformly Distributed Noise

Figure 5.26: Global Representation Components for D.E. for Hanging Initial State for Uniformly Distributed Noise
condition and uniformly distributed error and figure 5.28 shows the uncertainty ellipsoid size.

Figure 5.27: Error for the Hanging Initial Condition Using Uniformly Distributed Noise

Figure 5.28: Uncertainty for the Hanging Initial Condition Using Uniformly Distributed Noise
CHAPTER 6
CONCLUSIONS

This document summarizes the comparison of two state estimation schemes for a spherical pendulum. The two estimation schemes are the Extended Kalman Filter and a deterministic estimation scheme which uses a global description of the system states based on geometric mechanics. The comparison is meant to reduce the risk associated with using the deterministic estimation scheme for satellite attitude determination instead of the often used scheme of EKF. The spherical pendulum system is chosen as a simplified analog to a satellite. However, the angular velocities are much faster than satellite dynamics.

The deterministic estimation scheme uses uncertainty ellipsoids represented by uncertainty matrices that are propagated with the state estimate until a sensor reading is available. The sensor reading also has an uncertainty ellipsoid associated with it and the filter step is completed by fitting the smallest possible ellipsoid about the volume of intersection between the propagated and sensor ellipsoids. The deterministic estimate does not require any knowledge about the distribution of the uncertainty within the uncertainty ellipsoid, only that the uncertainty can be bounded by an ellipsoid. The state is represented, for the deterministic estimation scheme, by the reduced attitude matrix, $\Gamma$, and the angular velocity vector, $\omega$.

The Extended Kalman Filter uses the Kalman gain, calculated with the covariance of the propagated state estimate and the covariance of the sensor, to scale the innovation of the sensor measurement to create an updated state estimate. The state is represented for the EKF by spherical coordinates, $\theta$ and $\phi$ and angular velocity components $\omega_1$ and $\omega_2$.

The comparison shows that when the motion of the pendulum is localized around the stable equilibrium point, the hanging position, the EKF performs slightly better than the deterministic estimator. However when the motion includes the entire range by starting at the unstable equilibrium point, the inverted position, the deterministic estimation scheme performs better than the EKF. The comparison is also completed for Gaussian and uniformly
distributed noise. In this case the EKF results in a larger error compared to the performance of the deterministic estimator which does not result in error significantly different than that found in the Gaussian noise simulations. The EKF reliance on Gaussian noise models is a hindrance when dealing with uncharacterized noise.

In the future, it would be important to develop a Kalman-like filter that would utilize the global representation so as to completely eliminate the ambiguity issue. The comparison should also be carried out using other non-Gaussian noise distributions, including a noise model based on actual sensor noise. It would also be beneficial to incorporate model noise into the deterministic estimation scheme. A Monte-Carlo study of the results of final error would also provide valuable information on the sensitivity of the comparison to the initial conditions and sensor readings, especially for the inverted cases where results were found to vary between simulations. Repeating the comparison and including a particle filter could also be valuable.

The comparison shows that, in most situations, the deterministic estimation scheme performs as well as or better than the Extended Kalman Filter and contains no ambiguities. The deterministic estimation scheme should perform sufficiently well for the attitude determination of HawaiiSat-1.
BIBLIOGRAPHY


