University of Alberta

Implementation of a 6DOF Navigation System for an Unmanned Aerial Vehicle using an Extended Kalman Filter

by

Juan Carlos Fandino Ospina ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**

in

Controls

Department of Electrical and Computer Engineering

Edmonton, Alberta
Fall 2008

Your file Votre référence
ISBN: 978-0-494-47248-4
Our file Notre référence
ISBN: 978-0-494-47248-4

Abstract

This thesis treats the design and implementation of a 6 degree of freedom (6DOF) Navigation System which integrates an Inertial Measurement Unit (IMU), magnetometer, and Global Positioning System (GPS). Observability properties of the kinematics are investigated. The proposed navigation system has the property of being observable for any vehicle trajectory. Simulation and actual data is used to validate the proposed navigation algorithm.

Acknowledgements

I would like to thank my parents for supporting me in my decision of doing my Masters, for every long distance call they made to cheer me up and for being what they are, this work is especially dedicated to them. I would also like to thank Dr. Alan Lynch for giving me the opportunity to work with his ANCL Lab and for his cooperation as my supervisor in this research. Special thanks to David Kastelan for his dedication, cooperation, work ethic, professionalism and friendship. I have met wonderful colleagues at the ANCL Lab and which I would like to mention: David, Christopher, Karla, Thomas, Yebin, Martin, Edward, Rasoul, Kim, Katharina, Loren and Steve, I am thankful to you for making my Masters a joyful experience.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The Applied Nonlinear Controls Lab (ANCL) has an helicopter that will be operated as an Unmanned Aerial Vehicle (UAV). Several advantages can be found with these types of vehicles like vertical take-off, low speed flight and hover. Controlling this type of UAV is a challenging task because of its non-linear dynamics characteristics and underactuated behavior. In order to accomplish the task of flight control, it is necessary to implement a Navigation System that provides reliable navigation data.

Implementation of a Six Degree Of Freedom 6DOF Kalman Filter has been developed recently using Global Positioning System (GPS) as the only aiding measurement assuming the Inertial Measurement Unit (IMU) and the GPS sensors are placed at the same position [1]. The problems with this approach are:

1. The GPS antenna is not co-located with the IMU.

2. The UAV must undergo an observable trajectory and when it does not, unobservability causes the Extended Kalman Filter (EKF) estimates to diverge.

The work in [2] investigates the observability of an eighteen state navigation model with single antenna GPS position information. The eighteen states estimated in [2] are position, velocity, attitude, accelerometer bias, gyroscope bias and lever arm. It was proven that this model is observable if the vehicle is maneuvering.

This thesis will present a fifteen state scheme. It will not try to estimate the lever arm and will assume this vector is known in body frame coordinates. With the new set of measurements (position, acceleration and magnetic field), we will show that the kinematics will always be observable. Experimental testing will validate the proposed EKF scheme.

1.1 Thesis Contribution

In this thesis the design of an EKF using GPS, accelerometers and magnetometers measurements is investigated.

The main contributions of this thesis are:

- Model the effect of lever arm between the GPS antenna and IMU sensor. This is an improvement over the EKF implemented in [1].
- Investigate the observability when GPS position is the only aiding measurement. Previous work follows an eighteen state kinematic model [2]. Although the approach of investigating the observability properties is based on the latter reference, this thesis uses a fifteen state model and assumes the lever arm is known.
- Design of an EKF that uses Magnetometer/GPS/Accelerometer as sensor aiding measurements. This will ensure the system kinematics will always be observable.

1.2 Thesis Outline

- **Chapter 2** treats a three degree of freedom (3DOF) navigation problem and investigates the observability of the kinematic model with GPS position as the only aiding sensor.
- **Chapter 3** explains the six degrees of freedom (6DOF) kinematic problem. A brief introduction to sensor noise modeling is given. It also presents the estimation of the fifteen navigation variables and derives a linear error model that is used in the EKF.

- **Chapter 4** analyzes the observability of the 6DOF kinematic model for several UAV trajectories.

- **Chapter 5** discusses the inclusion of acceleration and magnetic fields as two extra aiding measurements for the EKF implementation. These added measurements makes the system observable.

- **Chapter 6** presents a comparison between single GPS-aided-INS and Magnetometer-GPS-aided-INS implementation with real data. It will explain the main advantage of using the magnetic and gravitational fields as extra measurements in the estimation of the navigation states.

- **Chapter 7** is the conclusion of this thesis. It will also provide potential future work.

Chapter 2

3DOF Kinematics

The vehicle can move in two dimensions on the horizontal plane. The x-axis points towards True North and y-axis towards True East. The UAV moves in the forward ($\hat{u}$) and rightward ($\hat{v}$) directions and rotates about the downward direction ($\hat{w}$). Accelerometers and gyroscopes measure the forward and rightward accelerations $\tilde{a}_u$, $\tilde{a}_v$ and the downward gyro rate $\tilde{\omega}_w$.

Downward acceleration ($\tilde{a}_w$) as well as forward and rightward gyro rotations ($\tilde{\omega}_u$, $\tilde{\omega}_v$) are movements that the UAV can not perform in a two dimensional plane. Thus, neither gravity nor earth rotation vectors are taken into account when analyzing the kinematics of a confined two-dimensional vehicle.

2.1 Reference Frames and Coordinate Systems

The two frames of interest, as explained in [3, Sec. 2] are:

- *Geographic Frame* (Earth Fixed Frame): The fixed frame that rotates with the earth. The fixed point is generally chosen at a known position in ECEF coordinates and taken as the origin of the Earth Fixed Frame Coordinate System. North, East and Down are the three orthogonal axes of the geographic frame.

- *Body Frame*: The frame that is fixed to the body. The origin is generally at the center of mass of the vehicle. Forward, Rightward and Down are the three orthogonal axes of this frame.

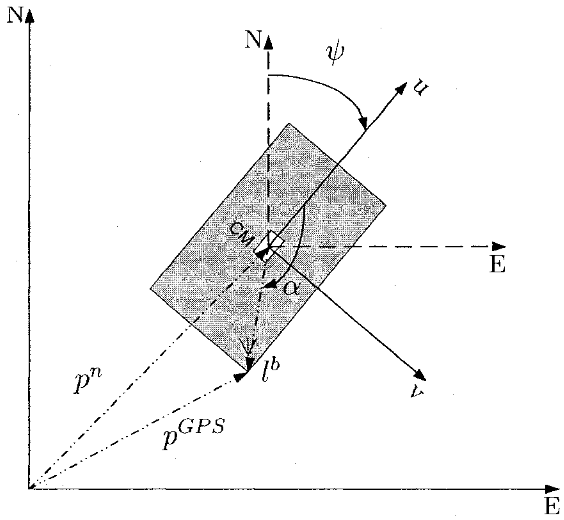


Figure 2.1: 2D Reference Frames

2.2 Two Dimensional Kinematic Equations

This section will explain the two dimensional kinematic model. Much of the material in this section is based on the work in [3].

2.2.1 Actual Kinematic Model

The component-wise equations for the 3DOF kinematic model are

$$\begin{bmatrix} \dot{p}_N(t) \\ \dot{p}_E(t) \\ \dot{v}_N(t) \\ \dot{v}_E(t) \\ \dot{\psi}(t) \end{bmatrix} = \begin{bmatrix} v_N(t) \\ v_E(t) \\ a_u(t)\cos\psi(t) - a_v(t)\sin\psi(t) \\ a_u(t)\sin\psi(t) + a_v(t)\cos\psi(t) \\ \omega_w(t) \end{bmatrix} \tag{2.1}$$

where p_N, p_E are north and east position components respectively, v_N, v_E are north and east velocity components respectively and ψ is the yaw or heading of the vehicle with respect to true north. a_u, a_v represent the forward and rightward acceleration components of the vehicle respectively and ω_w is the rotation rate of the vehicle about the downward axis.

The GPS antenna is placed off-center creating a lever arm between the center of mass (COM) of the vehicle and the GPS antenna. The actual GPS position equation in terms of the vehicle's COM position is

$$y = \begin{bmatrix} p_{GPS,N}(t) \\ \tilde{p}_{GPS,E}(t) \end{bmatrix} = \begin{bmatrix} p_N(t) + l_u\cos\psi(t) - l_v\sin\psi(t) \\ p_E(t) + l_u\sin\psi(t) + l_v\cos\psi(t) \end{bmatrix} \tag{2.2}$$

where $p_{GPS,N}, p_{GPS,E}$ are the north and east position components of the GPS antenna, p_N, p_E are the north and east position components of the vehicle's COM and l_u, l_v are the forward and rightward components of the lever arm vector l^b.

2.2.2 Introduction to Sensor Modeling

Acceleration a and rotation rate ω are measured by IMU sensors. These measurements are not perfect because they are usually corrupted by noise. Similarly GPS position p_{GPS} is measured by a GPS receiver which is also corrupted with noise.

The accelerometer measurement model used in this thesis is

$$\begin{bmatrix} a_u(t) \\ a_v(t) \end{bmatrix} = \begin{bmatrix} \tilde{a}_u(t) + b_a^u(t) + \nu_a^u(t) \\ \tilde{a}_v(t) + b_a^v(t) + \nu_a^v(t) \end{bmatrix} \tag{2.3}$$

$$\begin{bmatrix} \dot{b}_a^u(t) \\ \dot{b}_a^v(t) \end{bmatrix} = \begin{bmatrix} \nu_{b_a^u} \\ \nu_{b_a^v} \end{bmatrix} \tag{2.4}$$

where $\tilde{a}_u, \tilde{a}_v$ are the forward and rightward acceleration measurements respectively, b_a^u, b_a^v represent the bias in the forward and rightward direction of the accelerometers respectively, ν_a^u, ν_a^v represent the white noise disturbance for the forward and rightward accelerometers respectively and $\nu_{b_a^u}, \nu_{b_a^v}$ are the random walk noise model for the forward accelerometer bias and rightward accelerometer bias respectively.

The gyroscope measurement model used in this thesis is

$$\omega_w(t) = \tilde{\omega}_w(t) + b_\omega^w(t) + \nu_\omega^w(t) \tag{2.5}$$

$$\dot{b}_\omega^w = \nu_{b_\omega^w} \tag{2.6}$$

where $\tilde{\omega}^w$ is the rotation rate measurement about the downward axis, b_ω^w represents the bias in the downward direction of the gyroscope, ν_ω^w is the white noise disturbance of the downward gyroscope and $\nu_{b_\omega^w}$ is the random walk noise model for the downward axis gyroscope.

The GPS position measurement model used in this thesis is

$$\begin{bmatrix} \tilde{p}_{GPS,N} \\ \tilde{p}_{GPS,E} \end{bmatrix} = \begin{bmatrix} p_{GPS,N} \\ p_{GPS,E} \end{bmatrix} + \begin{bmatrix} \nu_N \\ \nu_E \end{bmatrix} \tag{2.7}$$

where $\tilde{p}_{GPS,N}, \tilde{p}_{GPS,E}$ represent the measurements of the north and east GPS antenna position, ν_N, ν_E represent the white noise disturbance of the position measurement in the north and east directions respectively.

2.2.3 Estimated Kinematic Model

The complete model of the vehicle's kinematics including the noise models presented in the last section is:

$$\begin{pmatrix} \dot{p}_N(t) \\ \dot{p}_E(t) \\ \dot{v}_N(t) \\ \dot{v}_E(t) \\ \dot{\psi}(t) \\ \dot{b}_a^u(t) \\ \dot{b}_a^v(t) \\ \dot{b}_\omega^w(t) \end{pmatrix} = \begin{pmatrix} v_N(t) \\ v_E(t) \\ (\tilde{a}_u + b_a^u)\cos\psi(t) - (\tilde{a}_v + b_a^v)\sin\psi(t) \\ (\tilde{a}_u + b_a^u)\sin\psi(t) + (\tilde{a}_v + b_a^v)\cos\psi(t) \\ \tilde{\omega}_w + b_\omega^w \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \nu_a^u \cos\psi(t) - \nu_a^v \sin\psi(t) \\ \nu_a^u \sin\psi(t) + \nu_a^v \cos\psi(t) \\ \nu_\omega^w \\ \nu_{b_a^u} \\ \nu_{b_a^v} \\ \nu_{b_\omega^w} \end{pmatrix} \tag{2.8}$$

The estimated version of (2.8) is:

$$\begin{pmatrix} \dot{\hat{p}}_N \\ \dot{\hat{p}}_E \\ \dot{\hat{v}}_N \\ \dot{\hat{v}}_E \\ \dot{\hat{\omega}} \\ \dot{\hat{b}}_a^u \\ \dot{\hat{b}}_a^v \\ \dot{\hat{b}}_\omega^w \end{pmatrix} = \begin{pmatrix} \hat{v}_N \\ \hat{v}_E \\ \left(\tilde{a}_u + \hat{b}_a^u\right)\cos\hat{\psi}(t) - \left(\tilde{a}_v + \hat{b}_a^v\right)\sin\hat{\psi}(t) \\ \left(\tilde{a}_u + \hat{b}_a^u\right)\sin\hat{\psi}(t) + \left(\tilde{a}_v + \hat{b}_a^v\right)\cos\hat{\psi}(t) \\ \tilde{\omega}_w + \hat{b}_\omega^w \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{2.9}$$

The complete model of the GPS antenna position including the noise model presented in the latter section is:

$$y = \begin{bmatrix} \tilde{p}_{GPS,N} \\ \tilde{p}_{GPS,E} \end{bmatrix} = \begin{bmatrix} p_N + l_u \cos\psi(t) - l_v \sin\psi(t) \\ p_E + l_u \sin\psi(t) + l_v \cos\psi(t) \end{bmatrix} + \begin{bmatrix} \nu_N \\ \nu_E \end{bmatrix} \tag{2.10}$$

The estimated version of (2.10) is:

$$\hat{y} = \begin{bmatrix} \hat{p}_{GPS,N} \\ \hat{p}_{GPS,E} \end{bmatrix} = \begin{bmatrix} \hat{p}_N + l_u \cos\hat{\psi}(t) - l_v \sin\hat{\psi}(t) \\ \hat{p}_E + l_u \sin\hat{\psi}(t) + l_v \cos\hat{\psi}(t) \end{bmatrix} \tag{2.11}$$

Where all "hat" symbols are estimates. Knowledge of lever arm vector $l^b = [l_u, l_v]$ in body frame coordinates is necessary. Reference [2] explains how they estimate the lever arm l^b increasing the order of the kinematic model to eighteen (i.e. $x \in \mathbb{R}^{18}$), however the vehicle must perform complicated maneuvers that are impossible to make in a three degree of freedom setup.

2.2.4 Linear Error Kinematic Model

Defining $x = [p_N, p_E, v_N, v_E, \psi, b_a^u, b_a^v, b_\omega^w]^T$ as the actual state, the estimated state as $\hat{x} = \left[\hat{p}_N, \hat{p}_E, \hat{v}_N, \hat{v}_E, \hat{\psi}, \hat{b}_a^u, \hat{b}_a^v, \hat{b}_\omega^w\right]$ and $\delta x = x - \hat{x} \in \mathbb{R}^8$ as the error state and subtracting (2.9) from (2.8), the kinematic state error to the first order is:

$$\begin{bmatrix} \delta\dot{p}_N \\ \delta\dot{p}_E \\ \delta\dot{v}_N \\ \delta\dot{v}_E \\ \delta\dot{\psi} \\ \delta\dot{b}_a^u \\ \delta\dot{b}_a^v \\ \delta\dot{b}_\omega^w \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\left(\tilde{a}_u + \hat{b}_a^u\right)\sin\psi - \left(\tilde{a}_v + \hat{b}_a^v\right)\cos\psi & \cos\psi & -\sin\psi & 0 \\ 0 & 0 & 0 & 0 & \left(\tilde{a}_u + \hat{b}_a^u\right)\cos\psi - \left(\tilde{a}_v + \hat{b}_a^v\right)\sin\psi & \sin\psi & \cos\psi & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta p_N \\ \delta p_E \\ \delta v_N \\ \delta v_E \\ \delta\psi \\ \delta b_a^u \\ \delta b_a^v \\ \delta b_\omega^w \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \cos\psi & -\sin\psi & 0 & 0 & 0 & 0 \\ \sin\psi & \cos\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \nu_a^u \\ \nu_a^v \\ \nu_\omega^w \\ \nu_{b_a^u} \\ \nu_{b_a^v} \\ \nu_{b_\omega^w} \end{bmatrix} \tag{2.12}$$

Defining $y = [\tilde{p}_{GPS,N}, \tilde{p}_{GPS,E}]^T$ as the actual output, $\hat{y} = [\hat{p}_{GPS,N}, \hat{p}_{GPS,E}]^T$ as the estimated output and $\delta y = y - \hat{y} \in \mathbb{R}^2$ as the output error and subtracting (2.11) from (2.10) the output error equation to the first order is:

$$\delta y = \begin{bmatrix} 1 & 0 & 0 & 0 & -(l_u \sin\psi + l_v \cos\psi) & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & (l_u \cos\psi - l_v \sin\psi) & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta p_N \\ \delta p_E \\ \delta v_N \\ \delta v_E \\ \delta\psi \\ \delta b_a^u \\ \delta b_a^v \\ \delta b_\omega^w \end{bmatrix} \tag{2.13}$$

Notice that the state, input and output matrices are dependent on ψ which is unknown.

2.3 3DOF Observability

We investigate the observability of the linear error kinematic model of the 3DOF system shown in the previous section (equations (2.12) and (2.13)) in order to find out whether a Kalman filter can be designed to estimate the UAV state totally or partially. If the system is observable partially, the class of trajectories for which the system is observable must be defined.

2.3.1 Observability: Constant Acceleration

Assuming the bias estimates and the sensor noise power spectral densities (PSD) are equal (i.e. $\tilde{a}_u + \hat{b}_a^u = a_u$ and $\tilde{a}_v + \hat{b}_a^v = a_v$), the non-zero entries of the observability matrix (defined in appendix B) of the system described by (2.12) and (2.13) are:

$$O = \begin{bmatrix} 1 & 0 & 0 & 0 & -(l_u \sin\psi + l_v \cos\psi) & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & (l_u \cos\psi - l_v \sin\psi) & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -(l_u \sin\psi + l_v \cos\psi) \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & (l_u \cos\psi - l_v \sin\psi) \\ 0 & 0 & 0 & 0 & -(a_u \sin\psi + a_v \cos\psi) & \cos\psi & -\sin\psi & 0 \\ 0 & 0 & 0 & 0 & (a_u \cos\psi - a_v \sin\psi) & \sin\psi & \cos\psi & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -(a_u \sin\psi + a_v \cos\psi) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & (a_u \cos\psi - a_v \sin\psi) \end{bmatrix} \tag{2.14}$$

The reduction of (2.14) into RREF gives the null space of the observability matrix $q(O)$ as:

$$q(O) = \left\{ \begin{bmatrix} \delta p_N \\ \delta p_E \\ \delta v_N \\ \delta v_E \\ \delta\psi \\ \delta b_a^u \\ \delta b_a^v \\ \delta b_\omega^w \end{bmatrix} : \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \frac{l_u \sin\psi + l_v \cos\psi}{a_u} & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & \frac{-l_u \cos\psi + l_v \sin\psi}{a_u} & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & \frac{1}{a_u} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & \frac{a_v}{a_u} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta p_N \\ \delta p_E \\ \delta v_N \\ \delta v_E \\ \delta\psi \\ \delta b_a^u \\ \delta b_a^v \\ \delta b_\omega^w \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right\} \tag{2.15}$$

The null space of O has dimension one if the vehicle's acceleration is constant. The subspace of the matrix affects the error states of position, attitude, and bias vectors. A basis for the null space of O is:

$$\text{span}\left\{\begin{bmatrix} -\frac{l_u \sin\psi + l_v \cos\psi}{a_u} \\ -\frac{-l_u \cos\psi + l_v \sin\psi}{a_u} \\ 0 \\ 0 \\ -\frac{1}{a_u} \\ -\frac{a_v}{a_u} \\ 1 \\ 0 \end{bmatrix}\right\} \tag{2.16}$$

Position, heading, forward accelerometer bias and rightward accelerometer bias error states will exhibit convergence problems while velocity and gyro bias error states will converge to zero. If the GPS antenna is placed exactly at the COM of the vehicle then the position error states will converge to zero, but the heading and rightward accelerometer bias error states will still exhibit convergence problems.

When the vehicle's acceleration is zero, the observability matrix O becomes

$$O = \begin{bmatrix} 1 & 0 & 0 & 0 & -(l_u \sin\psi + l_v \cos\psi) & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & (l_u \sin\psi - l_v \cos\psi) & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -(l_u \sin\psi + l_v \cos\psi) \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & (l_u \sin\psi - l_v \cos\psi) \\ 0 & 0 & 0 & 0 & 0 & \cos\psi & -\sin\psi & 0 \\ 0 & 0 & 0 & 0 & 0 & \sin\psi & \cos\psi & 0 \end{bmatrix} \tag{2.17}$$

The dimension of the null space of O is two. A basis of this null space is the span of the following vectors:

$$\text{span}\left\{\begin{bmatrix} (l_u \sin\psi + l_v \cos\psi) \\ -(l_u \sin\psi - l_v \cos\psi) \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ (l_u \sin\psi + l_v \cos\psi) \\ -(l_u \sin\psi - l_v \cos\psi) \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}\right\} \tag{2.18}$$

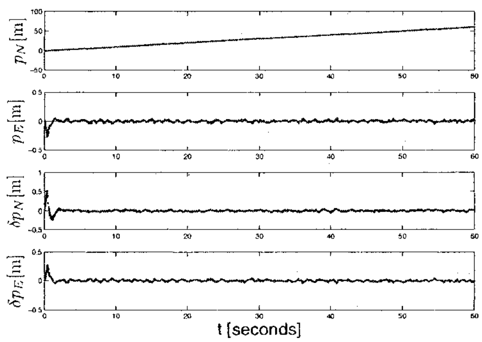
Accelerometer bias error states are the only error states that will converge to zero. The rest of the error states are unobservable in the direction of the null space. The heading and gyro bias errors affect, respectively, the position and velocity errors.

2.3.2 LTI Results and Interpretation

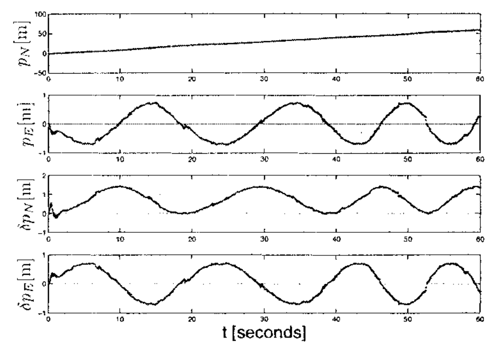
This section presents the simulation results for certain specific vehicle trajectories. The LTI simulations consider the cases in which:

1. The vehicle has constant velocity
2. The vehicle has constant acceleration.

In both scenarios, it is assumed the vehicle is aligned with true north and that the lever arm between IMU and GPS sensor is $l^b = [-0.7, 0]^T\,m$.

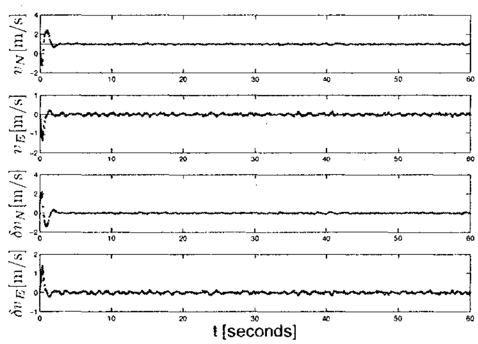


(a) Position without lever arm

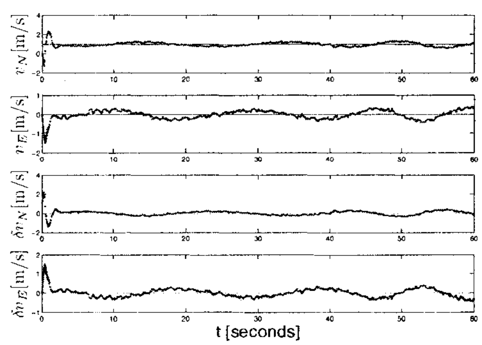


(b) Position with lever arm

Figure 2.2: Position Estimates and Errors

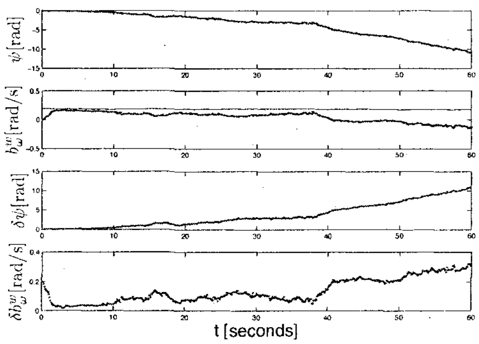


(a) Velocity without lever arm

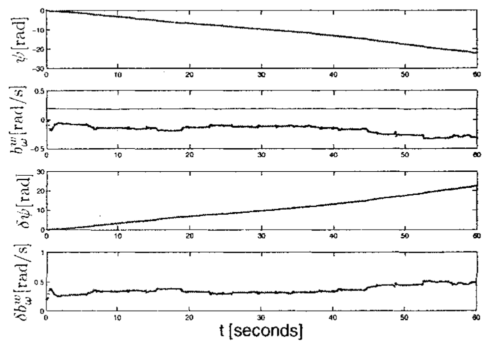


(b) Velocity with lever arm

Figure 2.3: Velocity Estimates and Errors

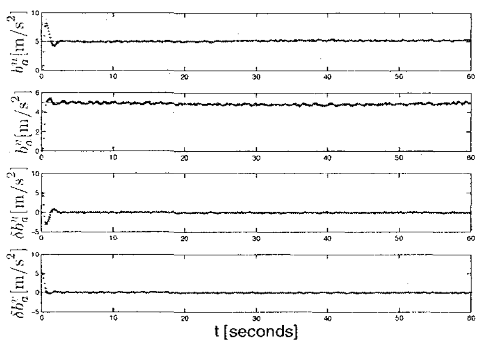


(a) Attitude and bias without lever arm

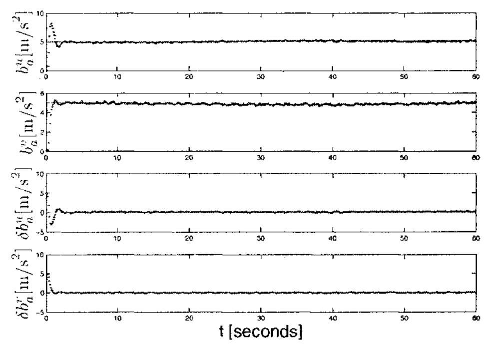


(b) Attitude and bias with lever arm

Figure 2.4: Attitude and Gyroscope bias Estimates and Errors



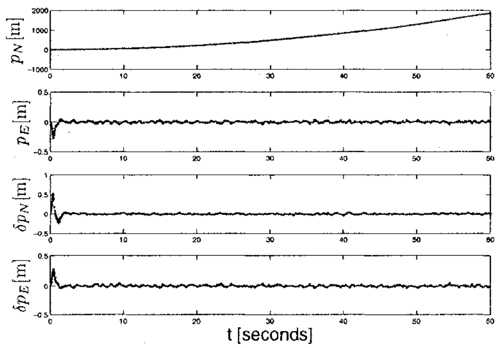
(a) Accelerometer bias without lever arm



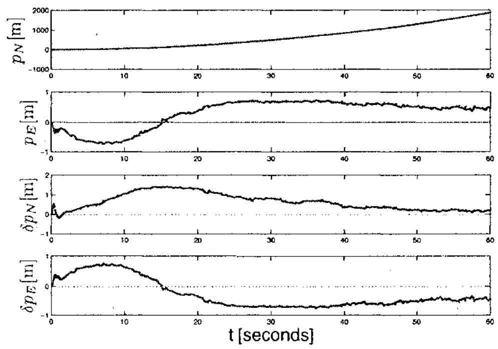
(b) Accelerometer bias with lever arm

Figure 2.5: Accelerometer bias Estimates and Errors

When the vehicle has zero acceleration and is aligned with true north, $\psi = 0$ and $[a_u, a_v] = [0, 0]$. The null space given in (2.18) suggests that only the accelerometer biases are going to be estimated correctly. Figures 2.2 to 2.5 display the results. Figure 2.5 indicates that the accelerometer biases converge to their actual values. Other error states are unobservable when there is a lever arm. Without the lever arm, position and velocity states will become observable as seen in Figures 2.2(a) and 2.3(a).

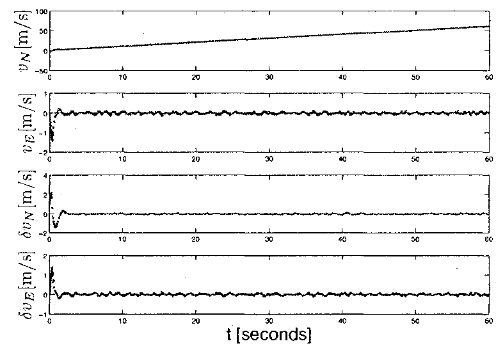


(a) Position without lever arm

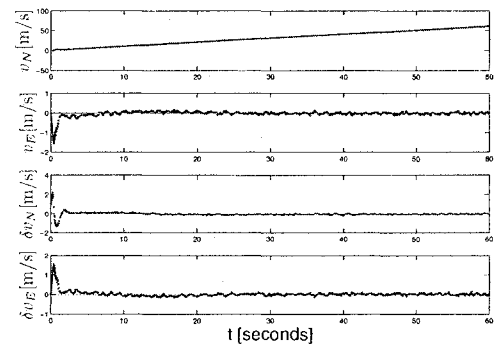


(b) Position with lever arm

Figure 2.6: Position Estimates and Errors

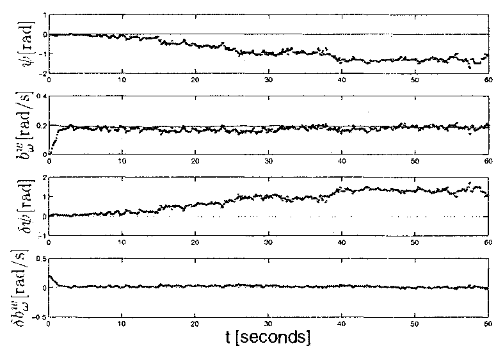


(a) Velocity without lever arm

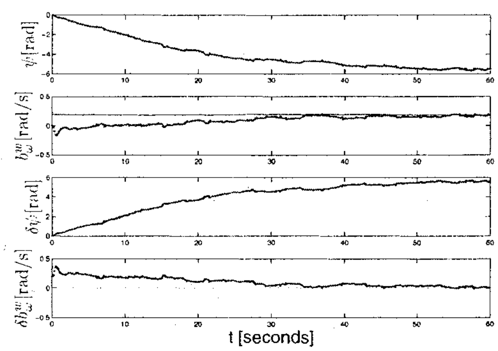


(b) Velocity with lever arm

Figure 2.7: Velocity Estimates and Errors



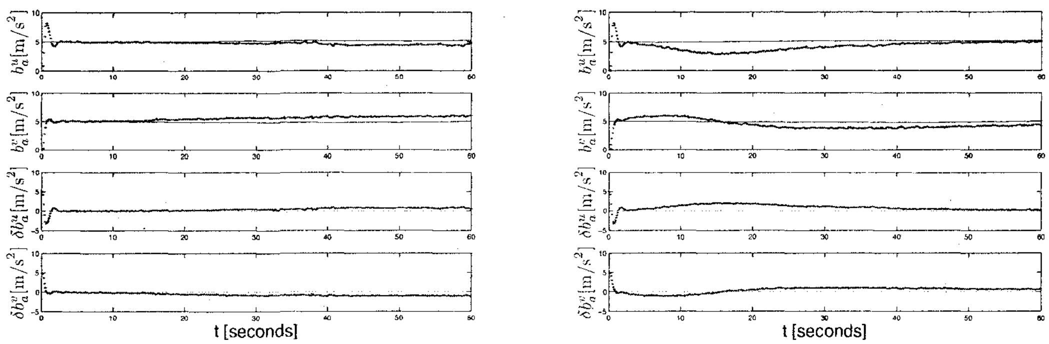
(a) Attitude and bias without lever arm



(b) Attitude and bias with lever arm

Figure 2.8: Attitude and Gyroscope bias Estimates and Errors

The simulation results given in figures 2.6 to 2.9 correspond to an acceleration in the north direction $\left([a_N; a_E] = [1, 0]\,\frac{m}{s^2}\right)$. The heading of the vehicle is



(a) Accelerometer bias without lever arm (b) Accelerometer bias with lever arm

Figure 2.9: Accelerometer bias Estimates and Errors

tangent to the direction of acceleration ($\psi = 0$). From (2.16), we know that the position, heading and accelerometer biases will have estimation errors, while the velocity and gyro bias states will be estimated correctly when the lever arm vector is considered. When the lever arm vector is zero, only the heading and accelerometer biases will have convergence problems.

When the vehicle is accelerating, the dimension of the null space is one, whereas when it is not, the null space dimension is two. It is reasonable to think that more states would have convergence problems with the two dimensional rather than the one dimensional null space. However, the null space basis forms a direction where the state space can not be estimated and will affect those states that are different from zero in the basis, regardless of the dimension of the null space. A bigger nullity would provide higher chances of more states having convergence problems. Nevertheless, there could be cases where a smaller nullity affects as many states as a bigger null space dimension.

If the vehicle is going at constant acceleration the null space dimension is one and the states affected are the position $\left([\delta p_N, \delta p_E]^T\right)$, heading $(\delta\psi)$ and acceleration biases $\left([\delta b_a^u, \delta b_a^v]^T\right)$. If the vehicle is non-accelerating, the null space dimension increases to two and the states affected are the position $\left([\delta p_N, \delta p_E]^T\right)$, velocity $\left([\delta v_N, \delta v_E]^T\right)$, heading $(\delta\psi)$ and gyro bias(δb_ω^w). Figures 2.6 to 2.9 present the results from the Kalman estimator without lever arm (left plots) and with lever arm (right plots).

2.3.3 Observability: Non Constant Acceleration

In this section we show that the linear kinematic error model can be made observable by maneuvering the vehicle. Letting $a(t) = -\left(l_u \sin\psi(t) + l_v \cos\psi(t)\right)$, $b(t) = \left(l_u \cos\psi(t) - l_v \sin\psi(t)\right)$, $c(t) = -\left(a_u(t)\sin\psi(t) + a_v(t)\cos\psi(t)\right)$ and $d(t) = \left(a_u(t)\cos\psi(t) - a_v(t)\sin\psi(t)\right)$, a truncated to the third order observability matrix $\left(\delta \overset{(3)}{y}\right)$ is:

$$O = \begin{bmatrix} 1 & 0 & 0 & 0 & a(t) & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & b(t) & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \dot{a}(t) & 0 & 0 & a(t) \\ 0 & 0 & 0 & 1 & \dot{b}(t) & 0 & 0 & b(t) \\ 0 & 0 & 0 & 0 & c(t) + \ddot{a}(t) & \cos\psi(t) & -\sin\psi(t) & 2\dot{a}(t) \\ 0 & 0 & 0 & 0 & d(t) + \ddot{b}(t) & \sin\psi(t) & \cos\psi(t) & 2\dot{b}(t) \\ 0 & 0 & 0 & 0 & \dot{c}(t) + \overset{(3)}{a}(t) & -\dot{\psi}(t)\sin\psi(t) & -\dot{\psi}(t)\cos\psi(t) & \dot{c}(t) + 3\ddot{a}(t) \\ 0 & 0 & 0 & 0 & \dot{d}(t) + \overset{(3)}{b}(t) & \dot{\psi}(t)\cos\psi(t) & -\dot{\psi}(t)\sin\psi(t) & \dot{d}(t) + 3\ddot{b}(t) \end{bmatrix}$$

A necessary condition for O to be of full column rank is that $c(t)$ and $d(t)$ must be differentiable. If the system is to be rendered unobservable and assuming $c(t)$ and $d(t)$ are differentiable, then $\ddot{a}(t) = -c(t)$, $\ddot{b}(t) = -d(t)$, $\overset{(3)}{a}(t) = -\dot{c}(t)$, $\overset{(3)}{b}(t) = -\dot{d}(t)$ and higher order derivatives of $a(t), b(t), c(t), d(t)$ must vanish.

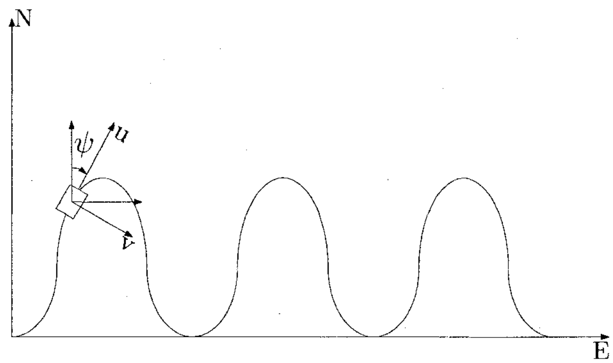


Figure 2.10: Vehicle 2D Trajectory

$$\begin{pmatrix} a_u \\ a_v \\ \psi \end{pmatrix} = \begin{pmatrix} a\frac{\pi^2}{25}\sin\left(\frac{\pi}{5}t + \frac{\pi}{2}\right) \\ 0 \\ \frac{a\frac{pi^2}{25}\sin\left(\frac{pi}{5}t+\frac{pi}{2}\right)}{\left(1+\left(\frac{pi}{5}a\cos\left(\frac{pi}{5}t+\frac{pi}{2}\right)\right)^2\right)} \end{pmatrix} \tag{2.19}$$

The trajectory shown in figure 2.10 is characterized by the acceleration and heading equations given in (2.19). The vehicle is moving in the east direction at constant velocity. With $l^b = [l_u, l_v] = [-0.7, 0]$ and $a = 2$ then:

$$a(t) = -l_u \sin\left(\frac{a\frac{\pi^2}{25}\sin\left(\frac{\pi}{5}t + \frac{\pi}{2}\right)}{1 + a^2\frac{\pi^2}{25}\left(\cos\left(\frac{\pi}{5} + \frac{\pi}{2}\right)\right)^2}\right) \tag{2.20}$$

$$b(t) = l_u \cos\left(\frac{a\frac{\pi^2}{25}\sin\left(\frac{\pi}{5}t + \frac{\pi}{2}\right)}{1 + a^2\frac{\pi^2}{25}\left(\cos\left(\frac{\pi}{5} + \frac{\pi}{2}\right)\right)^2}\right) \tag{2.21}$$

$$c(t) = -a\frac{\pi^2}{25}\sin\left(\frac{\pi}{5}t + \frac{\pi}{2}\right)\sin\left(\frac{a\frac{\pi^2}{25}\sin\left(\frac{\pi}{5}t + \frac{\pi}{2}\right)}{1 + a^2\frac{\pi^2}{25}\left(\cos\left(\frac{\pi}{5} + \frac{\pi}{2}\right)\right)^2}\right) \tag{2.22}$$

$$d(t) = a\frac{\pi^2}{25}\sin\left(\frac{\pi}{5}t + \frac{\pi}{2}\right)\cos\left(\frac{a\frac{\pi^2}{25}\sin\left(\frac{\pi}{5}t + \frac{\pi}{2}\right)}{1 + a^2\frac{\pi^2}{25}\left(\cos\left(\frac{\pi}{5} + \frac{\pi}{2}\right)\right)^2}\right) \tag{2.23}$$

It is obvious that $\ddot{a}(t) \neq -c(t)$, $\ddot{b}(t) \neq -d(t)$, $\overset{(3)}{a}(t) \neq -\dot{c}(t)$, $\overset{(3)}{b}(t) \neq -\dot{d}(t)$ and the higher order derivatives of $a(t)$, $b(t)$, $c(t)$ and $d(t)$ do not vanish. Therefore the trajectory given in (2.19) is observable and the EKF will estimate all states correctly.

Although all states can be estimated correctly, the lever arm vector decreases the rate of convergence of the EKF. In practice it is recommended to place the GPS antenna as close to the IMU as possible in order to improve the observability as long as it is the only aiding available.

Chapter 3

6DOF Kinematic Model

UAV are flying vehicles that have six degrees of freedom to move. Therefore UAV can translate in the forward, rightward and downward directions as well as rotate about the forward, rightward and downward directions ($[u, v, w]$).

3.1 Reference Frames and Coordinate Systems

The five reference frames of interest as explained in [1, Sec. 2] and [4, Ch. 2.1] are:

- *Inertial Frame*: This is a reference frame in which Newton's laws of motion apply. Therefore this frame must be non-accelerating. The vectors decomposed in this frame will have the following notation: $v^i = [v_x^i, v_y^i, v_z^i]$.

- *ECEF Frame*: Earth centered earth fixed. This frame is non inertial since the earth is rotating relative to the inertial frame and therefore accelerating. The origin is the geometric center of earth, Z-axis points to north, Y-axis points from the origin to 90^0 east of the greenwich meridian and lies at the equatorial plane, X-axis points from the origin to the intersection of the greenwich meridian with the equator. The vectors decomposed in this frame will have the following notation: $v^e = [v_x^e, v_y^e, v_z^e]$.

- *Geodetic Frame*: By default most GPS instruments report position in this coordinate system. The height h is the closest distance from the

UAV to the surface of earth. Note that the vector from the surface of earth to the UAV is normal to the surface of the earth; the latitude Φ is the angle formed by the extension of the height vector h to the equatorial plane; the longitude λ is the angle formed from the projection of the position vector from origin to UAV into the equatorial plane with the greenwich meridian.

- *Body Frame*: This frame is rigidly attached to the UAV. The origin is located at the COM of the UAV. The vectors decomposed in this frame will have the following notation: $v^b = [v_u, v_v, v_w]$ where u is the forward component, v is the rightward component and w is the downward component.

- *Navigation Frame*: All the data from the EKF is referenced to this frame. The origin is located at the ECEF position of the base station GPS antenna. The vectors decomposed in this frame will have the following notation: $v^n = [v_N, v_E, v_D]$ where N is the true north component, E is the true east component and D points to the inside of the surface of the earth.

3.2 Actual Kinematic Model

Assuming that the navigation frame:

1. Is the plane tangent to the position of the base station antenna.

2. Does not have motion relative to earth i.e. Ω_{en}^n=0.

The state equations for position, velocity and attitude are:

$$\begin{pmatrix} \dot{p}^n \\ \dot{v}^n \\ \dot{C}_b^n \end{pmatrix} = \begin{pmatrix} v^n \\ C_b^n f^b + g^n - (\Omega_{en}^n + 2\Omega_{ie}^n)\, v^n \\ C_b^n \Omega_{nb}^b \end{pmatrix} = \begin{pmatrix} v^n \\ C_b^n f^b + g^n - 2\Omega_{ie}^n v^n \\ C_b^n \Omega_{nb}^b \end{pmatrix} \quad (3.1)$$

where $p^n = [p_N, p_E, p_D]^T \in \mathbb{R}^3$ represent the position decomposed in navigation frame coordinates which are north, east and down components. $v^n = [v_N, v_E, v_D]^T \in \mathbb{R}^3$ is the velocity decomposed in navigation frame coordinates.

C_b^n is a Direction Cosine Matrix (DCM) that transforms vectors decomposed in body frame coordinates into navigation frame coordinates, this DCM is a function of the roll ϕ, pitch θ and yaw ψ. The vector ω_{ie}^n represents the relative angular velocity of the ECEF frame with respect to the inertial frame decomposed in navigation frame coordinates. The vector ω_{nb}^b represents the relative angular velocity of the body frame with respect to the navigation frame decomposed in body frame coordinates.

For computing the relative angular velocity from the body frame to the navigation frame (i.e. ω_{nb}^b), the following relation is useful [4, Ch. 6.2.2.3]:

$$\omega_{nb}^b = \omega_{ib}^b - C_n^b\left(\omega_{ie}^n + \omega_{en}^n\right) = \omega_{ib}^b - C_n^b\omega_{ie}^n \tag{3.2}$$

In equations (3.1) and (3.2), f^b is the specific force of the UAV decomposed body frame coordinates and ω_{ib}^b is the attitude rate of the body frame relative to the inertial frame decomposed in the body frame coordinates of the UAV.

The symbols Ω_{ie}^n and Ω_{nb}^b represent the skew symmetric matrix form of the vectors ω_{ie}^n and ω_{nb}^b respectively. They are usually used to represent the cross product of two vectors. For example:

$$\omega_{ie}^n \times v^n = \Omega_{ie}^n v^n = \begin{bmatrix} 0 & -\omega_{ie_D} & \omega_{ie_E} \\ \omega_{ie_D} & 0 & -\omega_{ie_N} \\ -\omega_{ie_E} & \omega_{ie_N} & 0 \end{bmatrix} \begin{bmatrix} v_N \\ v_E \\ v_D \end{bmatrix} \tag{3.3}$$

The DCM C_b^n as a function of ϕ, θ, ψ is:

$$C_b^n = \begin{bmatrix} \cos(\psi)\cos(\theta) & -\sin(\psi)\cos(\phi) + \cos(\psi)\sin(\theta)\sin(\phi) & \sin(\psi)\sin(\phi) + \cos(\psi)\sin(\theta)\cos(\phi) \\ \sin(\psi)\cos(\theta) & \cos(\psi)\cos(\phi) + \sin(\psi)\sin(\theta)\sin(\phi) & -\cos(\psi)\sin(\phi) + \sin(\psi)\sin(\theta)\cos(\phi) \\ -\sin(\theta) & \cos(\theta)\sin(\phi) & \cos(\theta)\cos(\phi) \end{bmatrix} \tag{3.4}$$

The time derivative of the DCM C_b^n given in [4, Ch. 2.5.1] is:

$$\dot{C}_b^n = C_b^n \Omega_{nb}^b \tag{3.5}$$

A further simplification can be made to the state space model (3.1) if we neglect the earth rotation (i.e. $\Omega_{ie}^n = 0$). The simplified model is:

$$\begin{pmatrix} \dot{p}^n \\ \dot{v}^n \\ \dot{C}_b^n \end{pmatrix} = \begin{pmatrix} v^n \\ C_b^n f^b + g^n \\ C_b^n \Omega_{ib}^b \end{pmatrix} \tag{3.6}$$

The main states that the navigation system is supposed to estimate are the position, velocity and attitude, the latter given in Euler angles relative to the navigation frame.

For the GPS position model, once the GPS base antenna's position $[\lambda_0, \Phi_0, h_0]$ is found (see Appendix D), then the origin of the navigation frame is defined and the relative position of the rover GPS antenna with respect to the base can be computed as follows:

$$p^n_{GPS} = C^n_e \left(p^e_{GPS_{rover}} - p^e_{GPS_{base}}\right) \tag{3.7}$$

where $p^e_{GPS_{base}}, p^e_{GPS_{rover}}$ are the position in ECEF coordinates of the base and the rover respectively. p^n_{GPS} represents the relative position between the rover with respect to the base decomposed in navigation frame. The DCM C^n_e is a function of λ_0 and Φ_0. It is computed as follows:

$$C^n_e = \begin{bmatrix} -\sin\lambda_0\cos\Phi_0 & -\sin\lambda_0\sin\Phi_0 & \cos\lambda_0 \\ -\sin\Phi_0 & \cos\Phi_0 & 0 \\ -\cos\lambda_0\cos\Phi_0 & -\cos\lambda_0\sin\Phi_0 & -\sin\lambda_0 \end{bmatrix} \tag{3.8}$$

The actual GPS position model in terms of the UAV's COM position is:

$$y = p^n_{GPS} = p^n + C^n_b l^b \tag{3.9}$$

where p^n_{GPS} is the GPS rover antenna's position decomposed in navigation frame coordinates and $l^b = [l_u, l_v, l_w]^T$ is the lever arm vector decomposed in body frame coordinates.

3.3 Sensors Models

It is necessary to estimate noise in the INS measurements by augmenting the kinematic model in (3.6) which includes specific force and gyro attitude biases.

IMU sensors measure the attitude rate and specific force in the body reference frame imperfectly. Thus, the measurements are modeled as their true values plus noise plus biases

$$\tilde{f}^b = f^b - \nu_a - b^b_a \tag{3.10}$$

$$\tilde{\omega}^b_{ib} = \omega^b_{ib} - \nu_g - b^b_\omega \tag{3.11}$$

where $\tilde{f}^b$ represents the measured specific force vector that is measured by the accelerometer sensors of the IMU, f^b represents the actual specific force vector. Since the IMU is assumed to be rigidly attached to the UAV, this vector is decomposed in body frame coordinates. The symbol ν_a represents white noise disturbance. The symbol b_a^b represents the bias disturbance of the accelerometers.

The symbol ω_{ib}^b represents the actual relative rotation of the body frame with respect to the navigation frame decomposed in body frame coordinates, $\tilde{\omega}_{ib}^b$ represents the measured relative rotation of the vector described before. The symbol ν_g represents the gyro white noise disturbance. The symbol b_ω^b represents the bias disturbance of the gyroscopes.

The specific force can be expressed as

$$f^b = \left(\frac{d^2 p^b}{dt^2}\right)_i + g^b \tag{3.12}$$

where p and g are position and gravitational field vectors respectively. Therefore, the IMU accelerometers besides measuring the UAV acceleration, they also measure the gravity vector. The addition of the acceleration and gravitational vectors is known as the specific force f.

Typically the bias noise components of the IMU sensors are modeled as Gauss-Markov processes:

$$\begin{bmatrix} \dot{b}_a^b \\ \dot{b}_\omega^b \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau_a} b_a^b + \nu_{b_a} \\ -\frac{1}{\tau_\omega} b_\omega^b + \nu_{b_\omega} \end{bmatrix} \tag{3.13}$$

where τ_a, τ_ω are the autocorrelation time constants of the accelerometers and gyroscopes respectively and $\nu_{b_a}, \nu_{b_\omega}$ are the driving white noise process for accelerometer and gyroscopes respectively. A further simplification to (3.13) can be made if we assume the time constants of the accelereometer and gyroscope biases are infinity (*i.e.* $\tau_a = \infty, \tau_\omega = \infty$). This will reduce the "Gauss-Markov process" to a "Random Walk process":

$$\begin{bmatrix} \dot{b}_a^b \\ \dot{b}_\omega^b \end{bmatrix} = \begin{bmatrix} \nu_{b_a} \\ \nu_{b_\omega} \end{bmatrix} \tag{3.14}$$

Similarly, GPS sensors measure position in ECEF coordinates imperfectly. These measurements can be transformed to navigation frame by means of the

DCM C_e^n. The GPS position solution error is proven to be bounded and thus there is no need to include a bias term as with INS. The GPS position model decomposed in navigation frame coordinates is:

$$\tilde{p}_{GPS}^n = p_{GPS}^n + \nu_{pGPS}^n \tag{3.15}$$

3.4 Estimated Kinematic Model

Keeping (3.6) and (3.14) in mind, define $\hat{x} = \left[\hat{p}^{n^T}, \hat{v}^{n^T}, \hat{C}_b^{n^T}, \hat{b}_a^{b^T}, \hat{b}_\omega^{b^T}\right]^T$ as the estimates of the state variables $x = \left[p^{n^T}, v^{n^T}, C_b^{n^T}, b_a^{b^T}, b_\omega^{b^T}\right]^T$. The complete model of the UAV's kinematics including the noise model presented in last section is

$$\begin{bmatrix} \dot{p}^n \\ \dot{v}^n \\ \dot{C}_b^n \\ \dot{b}_a^b \\ \dot{b}_\omega^b \end{bmatrix} = \begin{bmatrix} v^n \\ C_b^n f^b + g^n - 2\Omega_{ie}^n \\ C_b^n \Omega_{ib}^b \\ \nu_{b_a} \\ \nu_{b_\omega} \end{bmatrix} \tag{3.16}$$

The estimated version of (3.16) is:

$$\begin{bmatrix} \dot{\hat{p}}^n \\ \dot{\hat{v}}^n \\ \dot{\hat{C}}_b^n \\ \dot{\hat{b}}_a^b \\ \dot{\hat{b}}_\omega^b \end{bmatrix} = \begin{bmatrix} \hat{v}^n \\ \hat{C}_b^n \hat{f}^b + g^n \\ \hat{C}_b^n \hat{\Omega}_{ib}^b \\ 0 \\ 0 \end{bmatrix} \tag{3.17}$$

The complete output model including noise is

$$y = \tilde{p}_{GPS}^n = p^n + C_b^n l^b \tag{3.18}$$

The estimated version of (3.18) is

$$\hat{y} = \hat{p}_{GPS}^n = \hat{p}^n + \hat{C}_b^n l^b \tag{3.19}$$

where the lever arm l^b is assumed to be known.

3.5 First Order Error Equations

The error state $\delta x = x - \hat{x}$ is defined as

$$\begin{pmatrix} \delta p^n \\ \delta v^n \\ [\delta\rho\times] \\ \delta b_a^b \\ \delta b_\omega^b \end{pmatrix} = \begin{pmatrix} p^n - \hat{p}^n \\ v^n - \hat{v}^n \\ I - \hat{C}_b^n C_b^{n^T} \\ b_a^b - \hat{b}_a^b \\ b_\omega^b - \hat{b}_\omega^b \end{pmatrix} \tag{3.20}$$

The state equation for the error in position is

$$\delta\dot{p}^n = \dot{p}^n - \dot{\hat{p}}^n = v^n - \hat{v}^n = \delta v^n \tag{3.21}$$

The state equation for the error in velocity is

$$\delta\dot{v}^n = C_b^n f^b + g^n - \hat{C}_b^n \hat{f}^b - g^n = C_b^n \delta f^b - C_b^n f^b \times \delta\rho \tag{3.22}$$

The error in the rotation matrix estimation $\left(\text{i.e. } \hat{C}_b^n\right)$ is modeled as a skew symmetric matrix $P = [\delta\rho\times]$ as shown in equation (3.20) in the third row: $\hat{C}_b^n = (I - P)\, C_b^n$. A first-order dynamical model for the skew symmetric matrix P is

$$\dot{\hat{C}}_b^n = -\dot{P} C_b^n + (I - P)\, \dot{C}_b^n \tag{3.23}$$

$$\hat{C}_b^n \hat{\Omega}_{ib}^b = -\dot{P} C_b^n + (I - P)\, C_b^n \Omega_{ib}^b \tag{3.24}$$

$$\dot{P} C_b^n = C_b^n \Omega_{ib}^b - (I - P)\, C_b^n \hat{\Omega}_{ib}^b - P C_b^n \Omega_{ib}^b \tag{3.25}$$

$$\dot{P} = [\delta\dot{\rho}\times] = C_b^n \delta\Omega_{ib}^b C_b^{n^T} \tag{3.26}$$

Equation (3.26) expressed in vector form is

$$\delta\dot{\rho} = C_b^n \delta\omega_{ib}^b \tag{3.27}$$

The attitude error vector $\delta\rho$ is decomposed in navigation frame coordinates and represents misalignments between the axes of the true navigation frame with the computed navigation frame axes (see Figure 3.1). The following relations hold when the attitude error is small: $\delta\rho_N \approx \delta\phi$, $\delta\rho_E \approx \delta\theta$ and $\delta\rho_D \approx \delta\psi$.

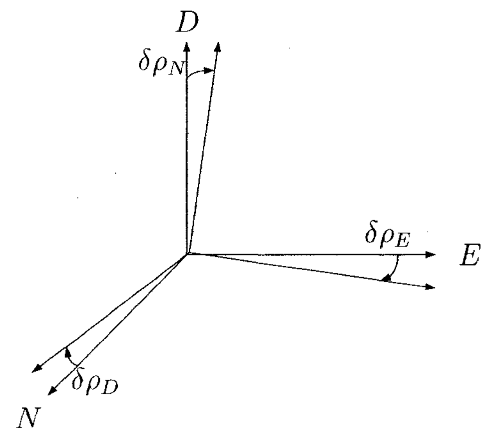


Figure 3.1: Attitude Error

The models for $\delta f^b, \delta\omega_{ib}^b, \hat{f}^b$ are not yet defined. Recalling equations (3.10) and (3.11) and defining their estimates as:

$$\hat{f}^b = \tilde{f}^b + \hat{b}_a^b \tag{3.28}$$

$$\hat{\omega}_{ib}^b = \tilde{\omega}_{ib}^b + \hat{b}_\omega^b \tag{3.29}$$

The sensor error models are:

$$\delta f^b = f^b - \hat{f}^b = \delta b_a^b + \nu_a \tag{3.30}$$

$$\delta\omega_{ib}^b = \omega_{ib}^b - \hat{\omega}_{ib}^b = \delta b_\omega^b + \nu_g \tag{3.31}$$

Recalling the last two rows of equation (3.20) and (3.14), the first-order dynamical model for the bias errors are:

$$\delta\dot{b}_a^b = \nu_{b_a} \tag{3.32}$$

$$\delta\dot{b}_\omega^b = \nu_{b_\omega} \tag{3.33}$$

Collecting equations (3.21), (3.22), (3.27), (3.32) and (3.33), in matrix form:

$$\begin{bmatrix} \delta\dot{p}^n \\ \delta\dot{v}^n \\ \delta\dot{\rho} \\ \delta\dot{b}_a^b \\ \delta\dot{b}_\omega^b \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & I & 0 & 0 & 0 \\ 0 & 0 & -\left[C_b^n f^b \times\right] & C_b^n & 0 \\ 0 & 0 & 0 & 0 & C_b^n \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta\rho \\ \delta b_a^b \\ \delta b_\omega^b \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ C_b^n & 0 & 0 & 0 \\ 0 & C_b^n & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix}}_{G} \begin{bmatrix} \nu_a \\ \nu_\omega \\ \nu_{b_a} \\ \nu_{b_\omega} \end{bmatrix} \tag{3.34}$$

The GPS position aiding measurement is the UAV's center of mass position plus the lever arm vector:

$$y = \tilde{p}^n_{GPS} = p^n + C^n_b l^b + \nu^n_{PGPS} \tag{3.35}$$

An estimated version of equation (3.35) is:

$$\hat{y} = \hat{p}^n + \hat{C}^n_b l^b \tag{3.36}$$

Subtracting (3.36) from (3.35), the following error output equation is obtained:

$$\delta y = \underbrace{\begin{bmatrix} I & 0 & -\left[C^n_b l^b \times\right] & 0 & 0 \end{bmatrix}}_{H} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta \rho \\ \delta b^b_a \\ \delta b^b_\omega \end{bmatrix} + \nu^n_{PGPS} \tag{3.37}$$

The mechanization equations that are going to be used for the EKF are:

- During the integration phase, the nonlinear update equation is:

$$\begin{bmatrix} \dot{\hat{p}}^n \\ \dot{\hat{v}}^n \\ \dot{\hat{C}}^n_b \\ \dot{\hat{b}}^b_a \\ \dot{\hat{b}}^b_\omega \end{bmatrix} = \begin{bmatrix} \hat{v}^n \\ \hat{C}^n_b \hat{f}^b + g^n - 2\Omega^n_{ie} \hat{v}^n \\ \hat{C}^n_b \hat{\Omega}^b_{nb} \\ 0 \\ 0 \end{bmatrix} \tag{3.38}$$

- During the measurement update phase, the linear state space model is:

$$\begin{bmatrix} \delta \dot{p}^n \\ \delta \dot{v}^n \\ \delta \dot{\rho} \\ \delta \dot{b}^b_a \\ \delta \dot{b}^b_\omega \end{bmatrix} = \begin{bmatrix} 0 & I & 0 & 0 & 0 \\ 0 & -2\Omega^n_{ie} & -\left[C^n_b f^b \times\right] & C^n_b & 0 \\ 0 & 0 & -\Omega^n_{ie} & 0 & C^n_b \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta \rho \\ \delta b^b_a \\ \delta b^b_\omega \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ C^n_b & 0 & 0 & 0 \\ 0 & C^n_b & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} \nu_a \\ \nu_\omega \\ \nu_{b_a} \\ \nu_{b_\omega} \end{bmatrix} \tag{3.39}$$

$$\delta y = \begin{bmatrix} I & 0 & -\left[C^n_b l^b \times\right] & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta \rho \\ \delta b^b_a \\ \delta b^b_\omega \end{bmatrix} + \nu^n_{PGPS} \tag{3.40}$$

When the lever arm is neglected ($l^b = 0$) then the third entry of equation (3.40) is zero.

Chapter 4

GPS aided INS Observability

Kalman Filtering is used in the integration of Inertial Navigation System (INS) and Global Positioning System (GPS) technologies to provide reliable navigation data. An INS uses a state space kinematic model which is integrated over time, GPS provides the system's measurements. The total state of the kinematic model must be inferred with GPS position aiding because the measurement dimensions ($\mathbb{R}^3$) is less than the system's state space order ($\mathbb{R}^{15}$). The states to be estimated are position, velocity, attitude, acceleration bias and gyroscope bias. With knowledge of the kinematic model, the sensors dynamical model and the GPS measurements, an EKF can be used to estimate the current state of a vehicle. Linear Observability analysis is necessary to determine if the estimator will converge and if so, for which vehicle trajectories.

4.1 UAV Observability Problem Statement

The observability of the linear kinematic error model of a 6DOF UAV is discussed in this section. The UAV that will be used in our practical analysis is a helicopter. The rotor of the helicopter is located at the vertical axis that passes through its COM, making it difficult to place the antenna along this same axis. The most viable position for the GPS antenna is the tail of the helicopter and the lever arm vector is generally expected to have components in the forward u, and downward w directions. Figure 4.1 shows the lever arm l^b between the UAV's COM and the GPS antenna.

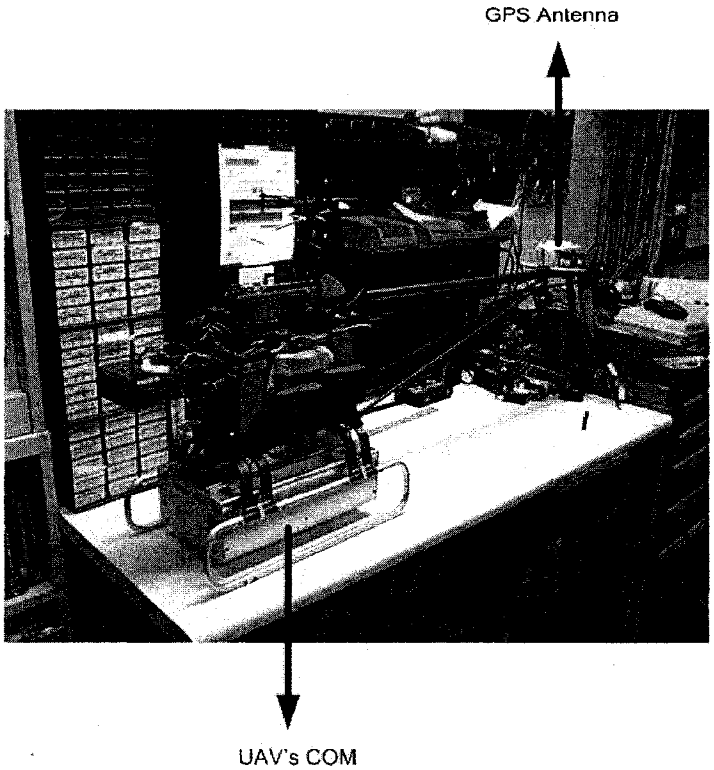


Figure 4.1: ANCL Helicopter

The state matrix is:

$$F(t) = \begin{bmatrix} 0 & I & 0 & 0 & 0 \\ 0 & 0 & -\left[C_b^n f^b \times\right] & C_b^n & 0 \\ 0 & 0 & 0 & 0 & C_b^n \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.1}$$

The output measurement matrix is:

$$H(t) = \begin{bmatrix} I & 0 & -\left[C_b^n l^b \times\right] & 0 & 0 \end{bmatrix} \tag{4.2}$$

The observability matrix O is a function of matrices F and H. Appendix B describes how to compute O from F and H. Three cases are analyzed:

1. The Linear Time Invariant (LTI) case where acceleration and attitude are constant.

2. The Linear Time Variant (LTV) case with variable acceleration and constant attitude.

3. The General LTV cases where acceleration and attitude of the vehicle changes.

4.2 LTI Observability Analysis

4.2.1 LTI Observability: Unbiased Measurements

Assuming all sensors are corrupted by white noise only, the biases are not taken into consideration and the linear error state equation becomes:

$$\begin{bmatrix} \delta\dot{p}^n \\ \delta\dot{v}^n \\ \delta\dot{\rho} \end{bmatrix} = \begin{bmatrix} 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & -\left[C_b^n f^b \times\right] \\ 0_3 & 0_3 & 0_3 \end{bmatrix} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta\rho \end{bmatrix} + \begin{bmatrix} 0_3 & 0_3 \\ C_b^n & 0_3 \\ 0_3 & C_b^n \end{bmatrix} \begin{bmatrix} \nu_a \\ \nu_\omega \end{bmatrix} \tag{4.3}$$

If the lever arm vector is not negligible, the output equation is:

$$\delta y = \begin{bmatrix} I_3 & 0_3 & -\left[C_b^n l^b \times\right] \end{bmatrix} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta\rho \end{bmatrix} \tag{4.4}$$

The observability matrix is

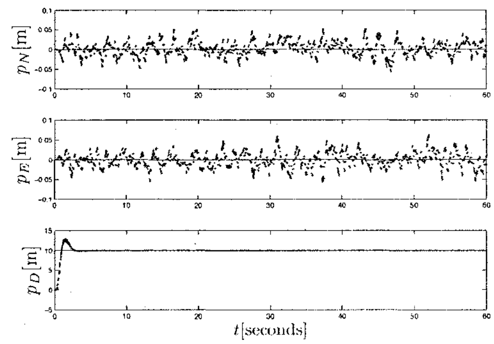
$$O = \left[\begin{array}{ccc} I_{3\times3} & 0_{3\times3} & -\left[C_b^n l^b \times\right] \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & -\left[C_b^n f^b \times\right] \\ \hline \multicolumn{3}{c}{0_{18\times9}} \end{array}\right] \tag{4.5}$$

The nullity of O is one independent of the value of l^b, and a null space basis $q(O)$ is:

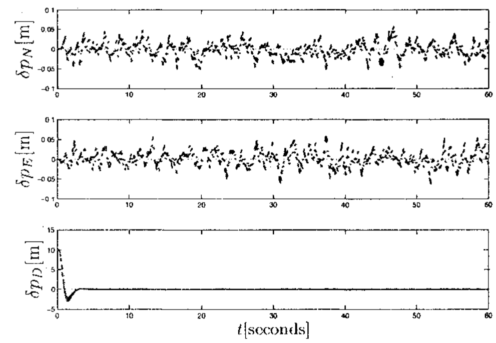
$$q(O) = \text{span}\left\{ \begin{bmatrix} l^n \times f^n \\ 0_{3\times1} \\ f^n \end{bmatrix} \right\} \tag{4.6}$$

Thus the position of the center of mass of the UAV becomes unobservable as the GPS antenna is displaced from the COM of the UAV. However, in case the specific force $f^n = C_b^n f^b$ and the lever arm $l^n = C_b^n l^b$ are aligned (i.e. $l^n = \alpha f^n$), the position error states become observable. This is only possible when the specific force is the gravity vector (i.e. $f^n = [0, 0, -g]^T$) and the UAV is level relative to the local tangent plane.

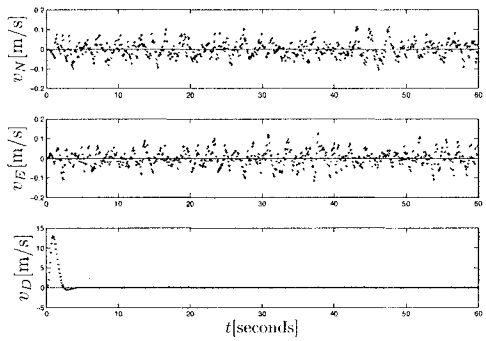
Figure 4.2 presents the simulation results for an EKF which estimates the UAV states assuming $l^b = 0$. The UAV is hovering at a height of ten meters above the origin, with a roll angle of zero ($\phi = 0$), a pitch angle of zero ($\theta = 0$)



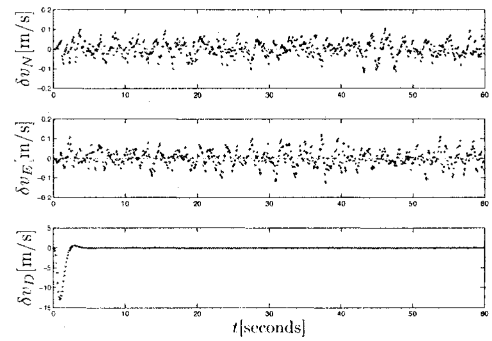
(a) Position states p^n



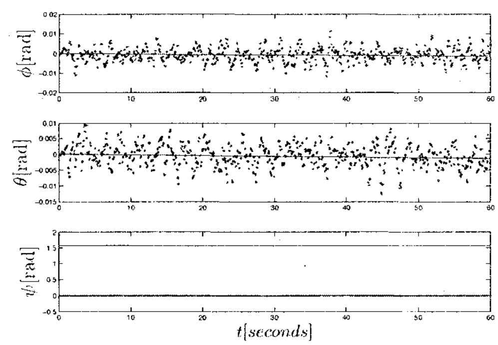
(b) Position error states δp^n



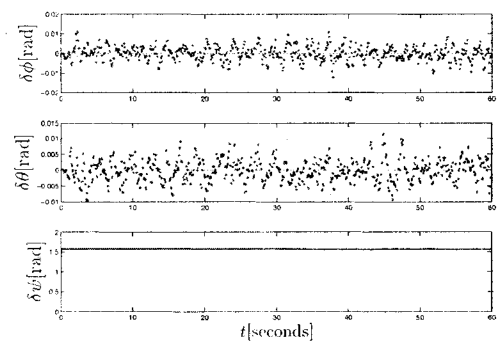
(c) Velocity states v^n



(d) Velocity error states δv^n

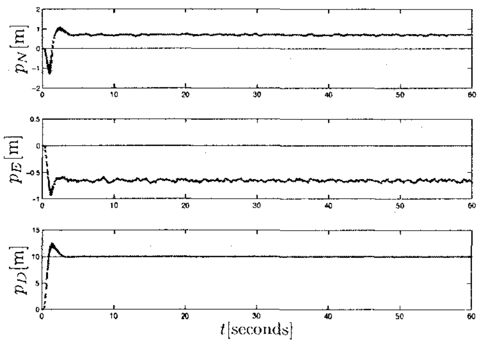


(e) Attitude states ϕ, θ, ψ

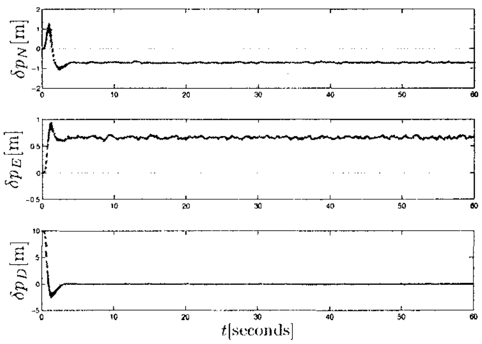


(f) Attitude error states $\delta\rho$

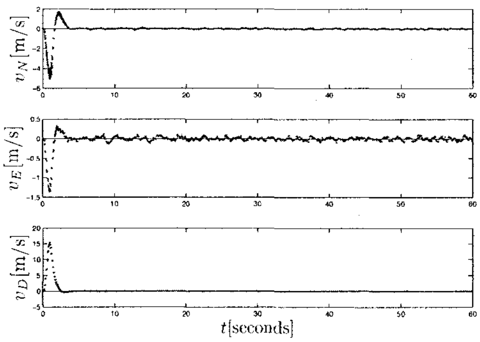
Figure 4.2: State Estimation: No Lever Arm



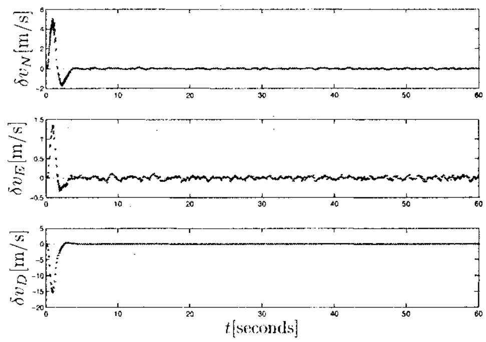
(a) Position States p^n



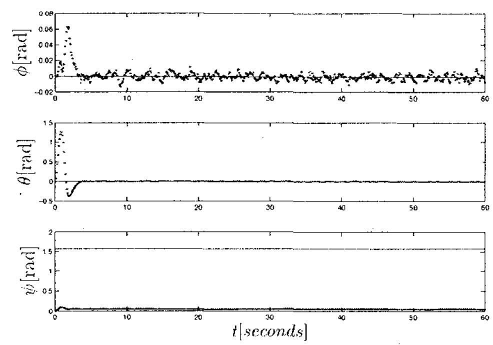
(b) Position error states δp^n



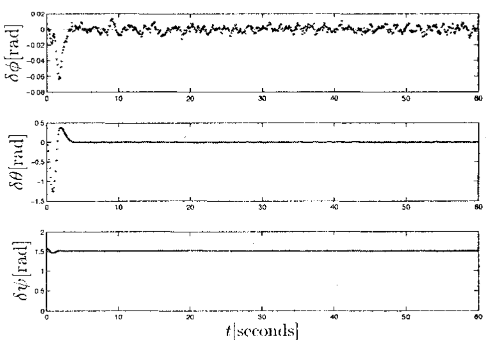
(c) Velocity states v^n



(d) Velocity error states δv^n



(e) Attitude states ϕ, θ, ψ



(f) Attitude error states $\delta\rho$

Figure 4.3: State Estimation: Lever Arm

and a heading angle different than zero $\left(\psi = \frac{\pi}{2}\right)$. Thus the UAV is aligned with true east hovering ten meters above ground.

Observing Subfigures 4.2(e) and 4.2(f), the yaw ψ state has convergence problems. Recalling that:

1. The null space (without the lever arm) of the unbiased system is in (4.6).

2. The accelerometers are measuring the gravity vector $\left(\text{i.e. } f^n = [0, 0, -g]^T\right)$.

3. The attitude error vector is: $[\delta\rho_N, \delta\rho_E, \delta\rho_D]^T \approx [\delta\phi, \delta\theta, \delta\psi]^T$.

the attitude in the direction of gravity is unobservable. Therefore, the heading or yaw angle ψ will be unobservable and its error is integrating the gyro sensor white noise (Angle Random Walk).

Figure 4.3 displays the results of the UAV state when a lever arm vector equal to $l^b = [-0.7, 0, 0]^T\ m$ is assumed. Furthermore, the heading of the UAV is towards east $\left(\psi = \frac{\pi}{2}\right)$, the vehicle is hovering ten meters above ground, and the vehicle has zero roll angle ($\phi = 0$) and zero pitch angle ($\theta = 0$).

Looking at Subfigures 4.3(e) and 4.3(f), it can be inferred that the yaw (ψ) has non zero steady state errors because the unobservable subspace is the span of $[l^n \times f^n, 0, f^n]^T$ and the specific force is $f^n = [0, 0, -g]^T$ affecting the heading state (ψ) only.

From Subfigures 4.3(a) and 4.3(b), it can be observed that position states has non zero steady state errors in the north (δp_N) and east (δp_E) directions. If the specific force vector is gravity $\left(\text{i.e. } f^n = [0, 0, -g]^T\right)$, then the yaw error state ($\delta\psi$) is unobservable, the roll and pitch error states ($\delta\phi, \delta\theta$) are observable, and the position measurement will be:

$$\delta y = \delta p^n - C_b^n l^b \times \delta\rho \tag{4.7}$$

using the lever arm vector $l^b = [l_u, l_v, 0]^T$ and $[\phi, \theta, \psi]^T = [0, 0, \frac{\pi}{2}]$, equation (4.7) becomes:

$$\delta y = \begin{bmatrix} \delta p_N \\ \delta p_E \\ \delta p_D \end{bmatrix} - \begin{bmatrix} 0 & 0 & l_u \\ 0 & 0 & l_v \\ -l_u & -l_v & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \delta\psi \end{bmatrix}$$

the observability of the north and east position error states ($\delta p_N, \delta p_E$) is affected. If the GPS antenna was placed at the vertical axis of the UAV and the UAV was hovering, then $l^n \times f^n = 0$ and the position error states (δp^n) would become observable.

4.2.2 Observability with Bias: Constant Acceleration

The linear system is described by (3.34) and (3.37) when the specific force f^b and rotation matrix C_b^n are kept constant. The observability matrix is:

$$O = \left[\begin{array}{ccccc} I_{3\times3} & 0_{3\times3} & -\left[C_b^n l^b \times\right] & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times3} & -\left[C_b^n l^b \times\right] C_b^n \\ 0_{3\times3} & 0_{3\times3} & -\left[C_b^n f^b \times\right] & C_b^n & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & -\left[C_b^n f^b \times\right] C_b^n \\ \hline \multicolumn{5}{c}{0_{33\times3}} \end{array}\right]_{45\times15} \tag{4.8}$$

The rank of O with or without the lever arm vector is eleven, and hence its nullity is four. Using $\delta x = \left[\delta p^{n^T}, \delta v^{n^T}, \delta \rho^T, \delta b_a^{b^T}, \delta b_\omega^{b^T}\right]^T \in \mathbb{R}^{15}$ as the state space coordinates, a basis for the null space $q(O)$ is:

$$q(O) = \text{span}\left\{ \begin{bmatrix} 0_{3\times1} \\ C_b^n\left(l^b \times f^b\right) \\ 0_{3\times1} \\ 0_{3\times1} \\ f^b \end{bmatrix}, \begin{bmatrix} [C_b^n l^b] \times \begin{bmatrix}0\\0\\1\end{bmatrix} \\ 0_{3\times1} \\ \begin{bmatrix}0\\0\\1\end{bmatrix} \\ C_b^{n^T}\left([C_b^n f^b] \times \begin{bmatrix}0\\0\\1\end{bmatrix}\right) \\ 0_{3\times1} \end{bmatrix}, \begin{bmatrix} [C_b^n l^b] \times \begin{bmatrix}0\\1\\0\end{bmatrix} \\ 0_{3\times1} \\ \begin{bmatrix}0\\1\\0\end{bmatrix} \\ C_b^{n^T}\left([C_b^n f^b] \times \begin{bmatrix}0\\1\\0\end{bmatrix}\right) \\ 0_{3\times1} \end{bmatrix}, \begin{bmatrix} [C_b^n l^b] \times \begin{bmatrix}1\\0\\0\end{bmatrix} \\ 0_{3\times1} \\ \begin{bmatrix}1\\0\\0\end{bmatrix} \\ C_b^{n^T}\left([C_b^n f^b] \times \begin{bmatrix}1\\0\\0\end{bmatrix}\right) \\ 0_{3\times1} \end{bmatrix} \right\} \tag{4.9}$$

The state coordinates δx have physical significance since they represent the error of the UAV's kinematics. Although the dimension of the unobservable subspace is unaffected by the value of the lever arm, the directions of unobservability is affected, and more states suffer convergence problems.

With no lever arm the null space basis in (4.9) becomes

$$\text{span}\left\{ \begin{bmatrix} 0_{3\times1} \\ 0_{3\times1} \\ 0_{3\times1} \\ 0_{3\times1} \\ f^b \end{bmatrix}, \begin{bmatrix} 0_{3\times1} \\ 0_{3\times1} \\ \begin{bmatrix}1\\0\\0\end{bmatrix} \\ C_b^{n^T}\left([C_b^n f^b] \times \begin{bmatrix}1\\0\\0\end{bmatrix}\right) \\ 0_{3\times1} \end{bmatrix}, \begin{bmatrix} 0_{3\times1} \\ 0_{3\times1} \\ \begin{bmatrix}0\\1\\0\end{bmatrix} \\ C_b^{n^T}\left([C_b^n f^b] \times \begin{bmatrix}0\\1\\0\end{bmatrix}\right) \\ 0_{3\times1} \end{bmatrix}, \begin{bmatrix} 0_{3\times1} \\ 0_{3\times1} \\ \begin{bmatrix}0\\0\\1\end{bmatrix} \\ C_b^{n^T}\left([C_b^n f^b] \times \begin{bmatrix}0\\0\\1\end{bmatrix}\right) \\ 0_{3\times1} \end{bmatrix} \right\} \tag{4.10}$$

The LTI observability conclusions are:

1. When $l^b = 0$, position and velocity errors δp^n, δv^n will always converge to zero according to the null space basis in (4.10). On the contrary when $l^b \neq 0$ position and velocity errors will exhibit convergence problems (see (4.9)).

2. The projection of the gyro bias error δb_ω^b into the specific force f^b vector is not observable. For example, when the vehicle is stationary, $f^b = [0, 0, -g]^T$. The down gyro bias component δb_{ww}^b will be unobservable.

3. The projection of the accelerometer bias δb_a^b into the cross product of specific force $f^n = C_b^n f^b$ with the attitude $\delta\rho$ of vehicle (i.e. $f^n \times \delta\rho$) when is rotated by $C_b^{n^T}$ is unobservable. For example, when the vehicle is hovering, $f^b = [0, 0, -g]^T$ and furthermore, letting all attitude angles be zero, $C_b^n = I_{3\times3}$. The only observable state is the bias in the down direction δb_{aw}^b.

State Space Decomposition

As a case study, we consider the case where the UAV is leveled relative to the navigation frame (i.e. $C_b^n = I$). The original state coordinates $\delta x = \left[\delta p^{n^T}, \delta v^{n^T}, \delta\rho^T, \delta b_a^{b^T}, \delta b_\omega^{b^T}\right]^T$ are going to be transformed into a new set of state coordinates $\delta\bar{x} = T\delta x$. The new state coordinates are divided into observable and unobservable subspace.

Theorem 6.O6 of [5] can be used in the transformation of the coordinates of the linear state space given by equations (3.34) and (3.37). This transformation divides the system into observable and unobservable subsystems. The theorem states that it is necessary to find and stack the linear independent (LI) rows of the observability matrix. The difference between the state space dimension and the number of LI rows defines the number of vectors that have to be picked arbitrarily in order to make the similarity transformation matrix T nonsingular. The following relations are useful:

$$\delta\bar{x} = T\delta x$$

$$\delta \dot{\bar{x}} = TFT^{-1}\delta\bar{x} + TB\delta u$$
$$\delta y = HT^{-1}\delta x$$

The dynamics $\delta \dot{x} = F\delta x$ is transformed into:

$$\begin{bmatrix} \delta\dot{\bar{x}}_O \\ \delta\dot{\bar{x}}_{NO} \end{bmatrix} = \begin{bmatrix} F_O & 0 \\ F_{21} & F_{NO} \end{bmatrix} \begin{bmatrix} \delta\bar{x}_O \\ \delta\bar{x}_{NO} \end{bmatrix} \tag{4.11}$$

It is necessary to find the matrix T that decomposes the system into observable and unobservable subsystems in order to determine the new set of states by the transformation $\delta\bar{x} = T\delta x$. Matrix T is attitude and acceleration dependent (C_b^n, f^n). In the case of hovering, it is assumed the UAV reaches equilibrium at a pitch angle (θ) of zero degrees and a roll angle (ϕ) of zero degrees. The yaw (ψ) does not affect the hovering of UAV and can assume any value. Two examples are presented:

1. The lever arm vector is equal to zero (i.e. $l^b = [0,0,0]^T$)
2. The lever arm vector is different than zero (i.e. $l^b = [l_u, 0, l_w]^T$).

Assuming the UAV is leveled, aligned and hovering, then $C_b^n = I_{3\times 3}$ and $f^n = f^b = [0,0,-g]^T$. Without lever arm, the coordinate transformation is

$$\delta\bar{x} = \left[\begin{array}{ccccccccccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & g & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & g & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -g & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & g & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g
\end{array}\right]
\begin{bmatrix} \delta p_N \\ \delta p_E \\ \delta p_D \\ \delta v_N \\ \delta v_E \\ \delta v_D \\ \delta\rho_N \\ \delta\rho_E \\ \delta\rho_D \\ \delta b^b_{au} \\ \delta b^b_{av} \\ \delta b^b_{aw} \\ \delta b^b_{\omega u} \\ \delta b^b_{\omega v} \\ \delta b^b_{\omega w} \end{bmatrix}
= \left[\begin{array}{c}
\delta p_N \\ \delta p_E \\ \delta p_D \\ \delta v_N \\ \delta v_E \\ \delta v_D \\ -g\delta\rho_E + \delta b^b_{au} \\ g\delta\rho_N + \delta b^b_{av} \\ \delta b^b_{aw} \\ -g\delta b^b_{\omega v} \\ g\delta b^b_{\omega u} \\ \hline \delta\rho_N - g\delta b^b_{av} \\ \delta\rho_E + g\delta b^b_{au} \\ \delta\rho_D \\ -g\delta b^b_{\omega w}
\end{array}\right] \tag{4.12}$$

Using a lever arm $l^b = [l_u, 0, l_w]$, the coordinate transformation is

$$\delta\bar{x} = \left[\begin{array}{ccccccccccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & l_w & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & -l_w & 0 & l_u & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & -l_u & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & l_w & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -l_w & 0 & l_u \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -l_u & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & g & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & g & 0 & 0 \\
\hline
0 & l_w & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -g & 0 & 0 & 0 & 0 \\
-l_w & 0 & l_u & 0 & 0 & 0 & 0 & 1 & 0 & g & 0 & 0 & 0 & 0 & 0 \\
0 & -l_u & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & gl_u & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g
\end{array}\right]
\begin{bmatrix} \delta p_N \\ \delta p_E \\ \delta p_D \\ \delta v_N \\ \delta v_E \\ \delta v_D \\ \delta\rho_N \\ \delta\rho_E \\ \delta\rho_D \\ \delta b^b_{au} \\ \delta b^b_{av} \\ \delta b^b_{aw} \\ \delta b^b_{\omega u} \\ \delta b^b_{\omega v} \\ \delta b^b_{\omega w} \end{bmatrix} \tag{4.13}$$

$$
\delta\bar{x} = \left[\begin{array}{c}
\delta p_N + l_w \delta\rho_E \\
\delta p_E - l_w \delta\rho_N + l_u \delta\rho_D \\
\delta p_D - l_u \delta\rho_E \\
\delta v_N + l_w \delta b^b_{\omega v} \\
\delta v_E - l_w \delta b^b_{\omega u} + l_u \delta b^b_{\omega w} \\
\delta v_D - l_u \delta b^b_{\omega v} \\
-g\delta\rho_E + \delta b^b_{au} \\
g\delta\rho_N + \delta b^b_{av} \\
\delta b^b_{aw} \\
-g\delta b^b_{\omega v} \\
g\delta b^b_{\omega u} \\
\hline
\delta\rho_N + l_w \delta p_E - g\delta b^b_{av} \\
-l_w \delta p_N + l_u \delta p_D + \delta\rho_E + g\delta b^b_{au} \\
-l_u \delta p_D + \delta\rho_D \\
g l_u \delta v_E - g\delta b^b_{\omega w}
\end{array}\right] \tag{4.14}
$$

Observe that position and velocity states are completely observable in the absence of the lever arm vector, while there is only a specific direction where these states are observable when the lever arm vector is considered.

In each case, the bottom four row vectors of the transformation matrix T belong to the null space of the observability matrices given in (4.8). To find a general expression for the transformation of the state space into observable and unobservable parts, it suffices to stack the first eleven LI row vectors of (4.8). The next four rows in these examples are taken from the basis of the null spaces (4.10) ($l^b = [0, 0, 0]^T$) and (4.9) ($l^b = [l_u, 0, l_w]^T$) respectively.

4.2.3 Eigenvalue Analysis

This section considers the stability of the unobservable modes in LTI. The only eigenvalue in the state matrix that is given in equation (3.34) for any attitude and acceleration is zero. This is because the state matrix has zeros on its diagonal and is upper triangular. The multiplicity of the zero eigenvalue is 15. The error state transition matrix transformed into Jordan form is

$$F_J = Q^{-1}FQ = \begin{bmatrix} 0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&1&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \end{bmatrix} \tag{4.15}$$

This Jordan form is not affected by the value of C_b^n and specific force vector $f^n = C_b^n f^b$. Solving $\delta\dot{\bar{x}} = F_J \delta\bar{x}$ gives:

$$\delta\bar{x}(t) = \begin{bmatrix} 1&t&\frac{t^2}{2!}&\frac{t^3}{3!}&0&0&0&0&0&0&0&0&0&0&0\\ 0&1&t&\frac{t^2}{2!}&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&1&t&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&1&t&\frac{t^2}{2!}&\frac{t^3}{3!}&0&0&0&0&0&0&0\\ 0&0&0&0&0&1&t&\frac{t^2}{2!}&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&t&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&t&\frac{t^2}{2!}&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&1&t&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&1&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&1&t&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \end{bmatrix} \delta\bar{x}(0) \tag{4.16}$$

As expected we observe certain error states may diverge from their true values. However, it is difficult to describe which of the original error states $\delta x = \left[\delta p^{n^T}, \delta v^{n^T}, \delta\rho^T, \delta b_a^{b^T}, \delta b_\omega^{b^T}\right]$ diverge since the new error state $\delta\bar{x} = Q\delta x$ is some linear combination of δx which depends on C_b^n and f^n.

Let's suppose that the UAV rests motionless on the ground and the body frame is leveled and aligned with navigation frame. In this state $C_b^n = I$. Transforming the state matrix into Jordan form as in (4.15), the coordinates are changed into $\delta\bar{x} = Q^{-1}\delta x$. Then, the new system coordinates $\delta\bar{x}$ are:

$$Q^{-1}\delta x = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{g} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{g} \\ 0 & 0 & \frac{1}{g} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{g} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{g} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{g} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}^{-1} \delta x = \begin{pmatrix} \delta p_N \\ \delta v_N \\ \delta b^b_{au} + g\delta\rho_E \\ g\delta b^b_{\omega v} \\ \delta p_E \\ \delta v_E \\ \delta b^b_{av} - g\delta\rho_N \\ -g\delta b^b_{\omega u} \\ \delta p_D \\ \delta v_D \\ \delta b^b_{aw} \\ \delta\rho_D \\ \delta b^b_{\omega w} \\ \delta b^b_{au} \\ \delta b^b_{av} \end{pmatrix} = \delta\bar{x} \tag{4.17}$$

From the homogeneous solution (4.16) and the new coordinates (4.17), we conclude $\delta b^b_{\omega v}$, $\delta b^b_{\omega u}$, δb^b_{aw}, $\delta b^b_{\omega w}$, δb^b_{av}, and δb^b_{au} will in general exhibit nonzero steady state values. Nine of the transformed states will in general diverge. We observe that δp_N and δv_N are unstable. It can be inferred that $\delta\rho_E \approx \delta\theta$, δp_E, δv_E, $\delta\rho_N \approx \delta\phi$, δp_D, δv_D, and $\delta\rho_D \approx \delta\psi$ are not convergent.

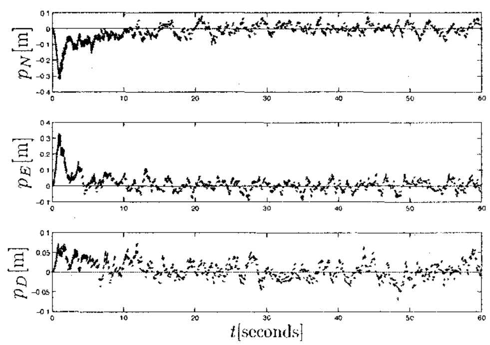
The transformation given in (4.17) works only when the body frame is leveled and aligned with the navigation frame and there is no acceleration other than gravity. As discussed before, the transformation is dependent on the rotation matrix C^n_b and the acceleration vector f^n and more complicated expressions may be found when $C^n_b \neq I$.

4.2.4 Simulation and Physical Interpretation

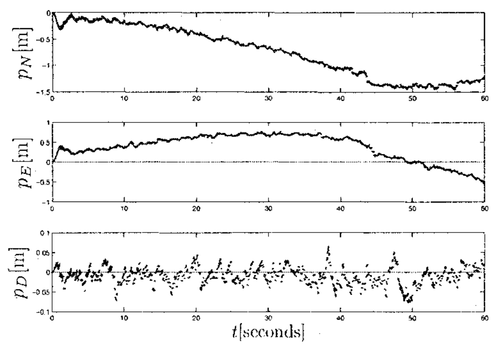
We Simulate an EKF with and without lever arm with a stationary UAV in order to check the observability of the states. We assume

1. The vehicle is located at the origin ($p^n = 0$)
2. The vehicle is leveled with the local tangent plane ($(\phi, \theta) = (0, 0)$).
3. The vehicle is aligned with true north ($\psi = 0$).

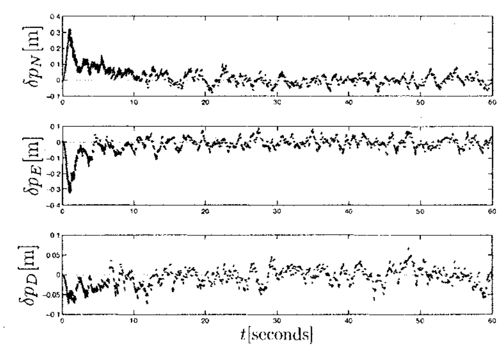
Looking at the left portions of figures 4.4 to 4.8, we see that, when lever arm is zero, the position error states δp^n, velocity error states δv^n, downward accelerometer bias error state δb^b_{aw}, forward gyro bias error state $\delta b^b_{\omega u}$ and rightward gyro bias error state $\delta b^b_{\omega v}$ converge since these states are completely



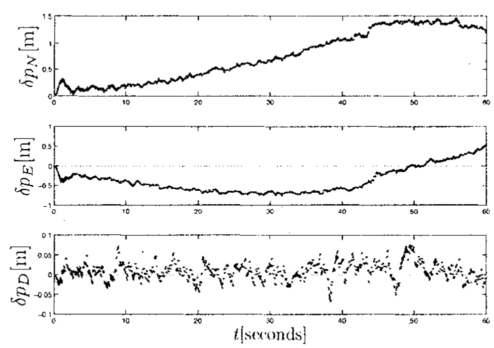
(a) Position states p^n



(b) Position states p^n: lever arm

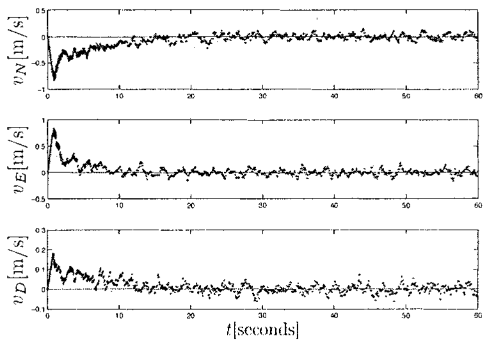


(c) Position error states δp^n

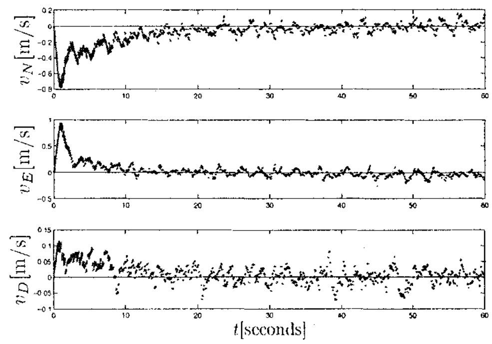


(d) Position error states δp^n: lever arm

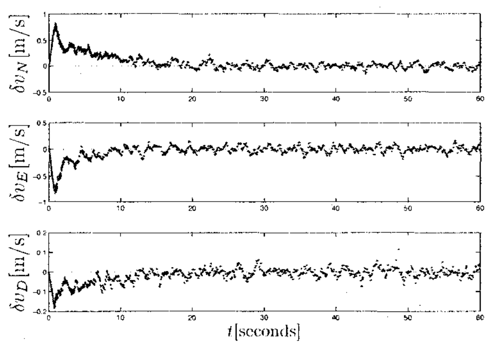
Figure 4.4: Position Estimation



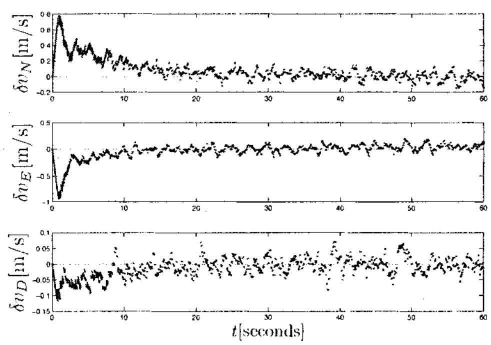
(a) Velocity states v^n



(b) Velocity states v^n: lever arm

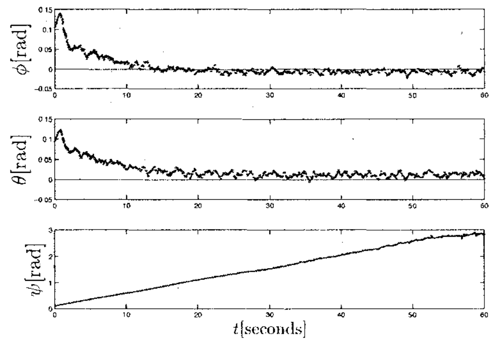


(c) Velocity error states δv^n

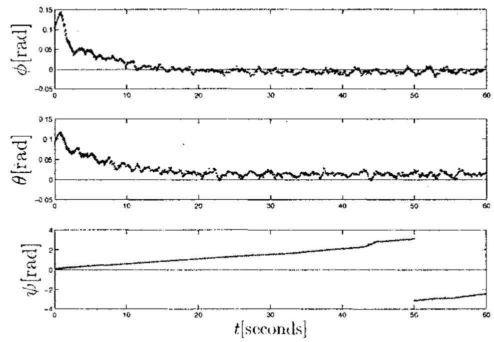


(d) Velocity error states δv^n: lever arm

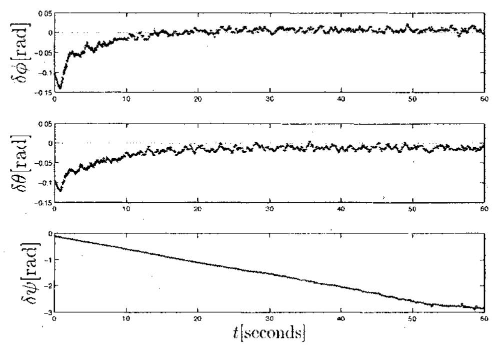
Figure 4.5: Velocity Estimation



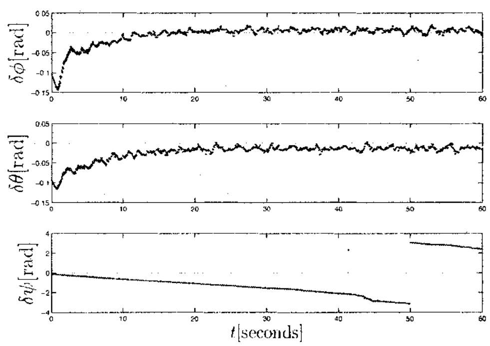
(a) Attitude states ϕ, θ, ψ



(b) Attitude states ϕ, θ, ψ: lever arm

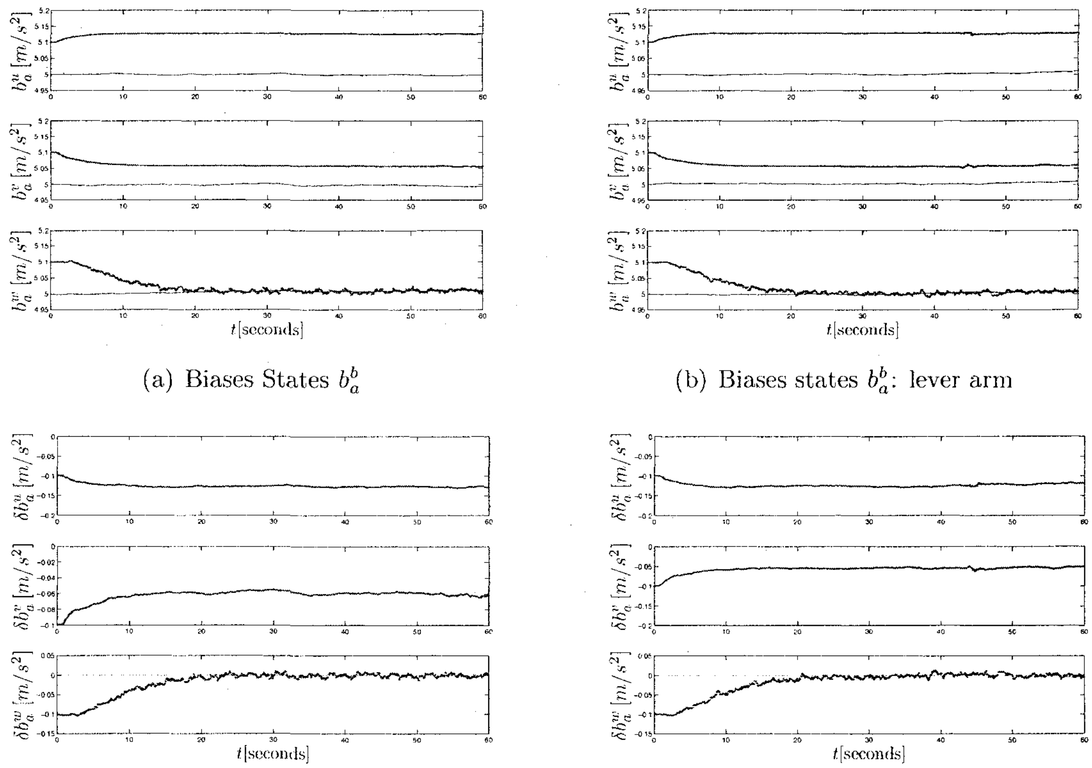


(c) Attitude error states $\delta\rho$



(d) Attitude error states $\delta\rho$: lever arm

Figure 4.6: Attitude Estimation



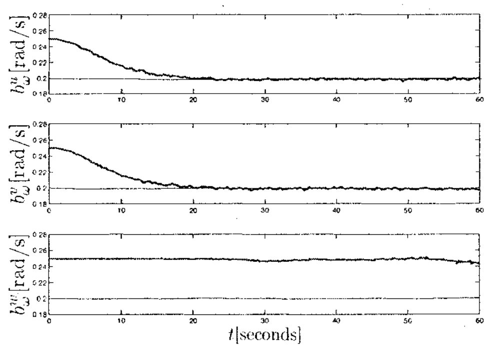
(a) Biases States b_a^b

(b) Biases states b_a^b: lever arm

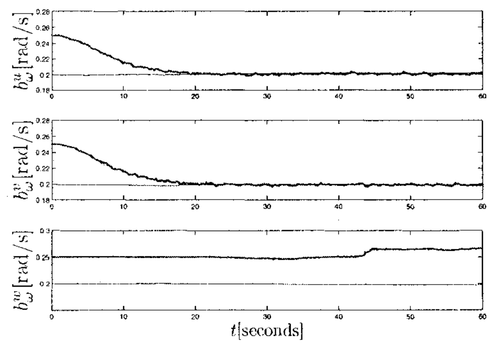
(c) Biases error states δb_a^b

(d) Biases error states δb_a^b: lever arm

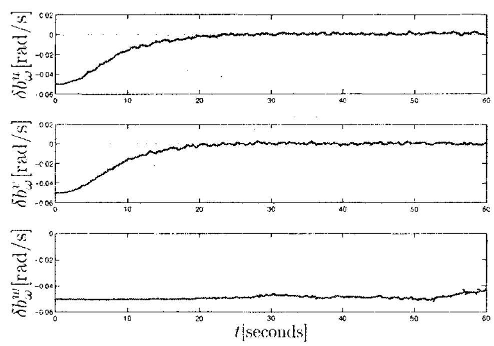
Figure 4.7: Acceleration Biases Estimation



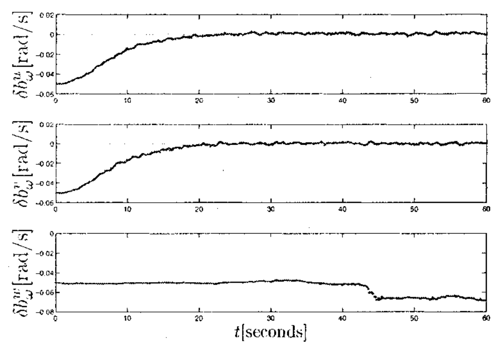
(a) Biases states b^b_ω



(b) Biases states b^b_ω: lever arm



(c) Biases error states δb^b_ω



(d) Biases error states δb^b_ω: lever arm

Figure 4.8: Gyroscope Biases Estimation

observable. On the other hand, the right portions of these figures show that convergence problems exist for all states except with the forward and rightward gyro bias error states $\delta b^b_{\omega u}, \delta b^b_{\omega v}$, the downward accelerometer bias error state $\left(\delta b^b_{aw}\right)$, downward position and velocity error states $\delta p_D, \delta v_D$ and north velocity error state δv_N.

4.3 LTV Observability: Zero Bias

This is the most trivial case of LTV observability analysis. Recalling the linear nine state space equation:

$$\begin{bmatrix} \delta \dot{p}^n \\ \delta \dot{v}^n \\ \delta \dot{\rho} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & -[f^n \times] \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta \rho \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ C^n_b & 0 \\ 0 & C^n_b \end{bmatrix} \begin{bmatrix} \nu_a \\ \nu_\omega \end{bmatrix}$$

$$\delta y = \begin{bmatrix} I & 0 & -[l^n \times] \end{bmatrix} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta \rho \end{bmatrix}$$

The observability test is performed on

$$O = \begin{bmatrix} I & 0 & -[l^n \times] \\ 0 & I & -\left[\dot{l}^n \times\right] \\ 0 & 0 & -[f^n \times] - \left[\ddot{l}^n \times\right] \\ 0 & 0 & -\left[\dot{f}^n \times\right] - \left[\overset{(3)}{l^n} \times\right] \\ \vdots & \vdots & \vdots \\ 0 & 0 & -\left[\overset{(n-3)}{f^n} \times\right] - \left[\overset{(n-1)}{l^n} \times\right] \end{bmatrix}$$

The system is observable if there are at least two LI vectors from the set $\left\{ f^n + \ddot{l}^n, \dot{f}^n + \overset{(3)}{l^n}, ..., \overset{(5)}{f^n} + \overset{(7)}{l^n} \right\}$.

Rotation alone can produce a time varying lever arm (e.g. $\dot{l}^n = C^n_b \Omega^b_{nb} l^b$ and so on). The time derivatives of the lever arm might be in the opposite direction to the specific force time derivatives (i.e. $f^n + \ddot{l}^n = 0$ and so on) and the nullity dimension might be as high as three.

4.4 LTV Observability with Bias: Variable Acceleration

In this section we assume variable f^b and constant C_b^n. The following fifteen-state observability matrix is obtained:

$$O = \begin{bmatrix} I & 0 & -\left[C_b^n l^b \times\right] & 0 & 0 \\ 0 & I & 0 & 0 & -\left[C_b^n l^b \times\right] C_b^n \\ 0 & 0 & -\left[C_b^n f^b \times\right] & C_b^n & 0 \\ 0 & 0 & -\left[C_b^n \dot{f}^b \times\right] & 0 & -\left[C_b^n f^b \times\right] C_b^n \\ 0 & 0 & -\left[C_b^n \ddot{f}^b \times\right] & 0 & -2\left[C_b^n \dot{f}^b \times\right] C_b^n \\ 0 & 0 & -\left[C_b^n \overset{(3)}{f^b} \times\right] & 0 & -3\left[C_b^n \ddot{f}^b \times\right] C_b^n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & -\left[C_b^n \overset{(12)}{f^b}\right] & 0 & -12\left[C_b^n \overset{(11)}{f^b} \times\right] C_b^n \end{bmatrix} \tag{4.18}$$

The first three block rows of matrix (4.18) are almost in row echelon form (REF). Observe also that the first two block rows of O will always provide six LI rows regardless of the value of the lever arm. The addition of the lever arm vector when there is no rotation does not affect the rank of the matrix in (4.18). Thus it suffices to check for the rank of the block submatrix at the bottom (below the first two block rows and to the right of the first two block columns). Three cases are the most important in the analysis of the rank of O:

1. When for all t, f^b and all its time derivatives are linearly dependent,
2. When for all t, f^b is LI with one of its time derivatives,
3. When for all t, f^b is LI with at least two of its time derivatives.

4.4.1 f^b and Time Derivatives are Linearly Dependent

It suffices to check that the rank of the block sub matrix at the bottom of (4.18) is full column rank:

$$\begin{bmatrix} -\left[C_b^n f^b \times\right] & C_b^n & 0 \\ -\left[C_b^n \dot{f}^b \times\right] & 0 & -\left[C_b^n f^b \times\right] C_b^n \\ -\left[C_b^n \ddot{f}^b \times\right] & 0 & -2\left[C_b^n \dot{f}^b \times\right] C_b^n \\ \vdots & \vdots & \vdots \\ -\left[C_b^n \overset{(12)}{f^b} \times\right] & 0 & -12\left[C_b^n \overset{(11)}{f^b} \times\right] C_b^n \end{bmatrix} \tag{4.19}$$

In order to do so, it is necessary to reduce the matrix to an almost REF. It is assumed that f^b is smooth and continuously differentiable at least to 12th order.

When all the components of the set $\left\{f^b, \dot{f}^b, ..., \overset{(12)}{f^b}\right\}$ are linearly dependent, we can express all the terms as a linear combination of one of the components. Any vector valued function picked from the above set is thus a basis that spans this set. Letting f^b be the basis of the set $\left\{f^b, \dot{f}^b, ..., \overset{(12)}{f^b}\right\}$, then by assumption all the vectors of the former set are in the same direction of f^b. Thus, the set transforms to $\left\{f^b, k_1 f^b, k_2 f^b, ..., k_{12} f^b\right\}$. The REF of (4.19) is:

$$\begin{bmatrix} -\left[C_b^n f^b \times\right] & C_b^n & 0 \\ 0 & -k_1 C_b^n & -\left[C_b^n f^b \times\right] C_b^n \\ 0 & 0 & -\left(2k_1^2 - k_2\right)\left[C_b^n f^b \times\right] C_b^n \\ 0 & 0 & -\left(3k_1 k_2 - k_3\right)\left[C_b^n f^b \times\right] C_b^n \\ \vdots & \vdots & \vdots \\ 0 & 0 & -\left(12k_1 k_{11} - k_{12}\right)\left[C_b^n f^b \times\right] C_b^n \end{bmatrix} \tag{4.20}$$

The first column of (4.20) factors attitude error states $\delta\rho$. The second column of (4.20) factors accelerometer biases error states δb_a^b. The last column of (4.20) factors gyro biases error states δb_ω^b.

Case 1 When any or all of $ik_1k_{i-1} - k_i \neq 0$, the nullity of (4.20) is 2. The null space is then spanned by two LI vectors in $\mathbb{R}^9$. The observability matrix in REF (matrix (4.20)) can be reduced to:

$$\begin{bmatrix} -\left[C_b^n f^b \times\right] & 0 & 0 \\ 0 & -k_1 C_b^n & 0 \\ 0 & 0 & -l_1\left[C_b^n f^b \times\right] C_b^n \end{bmatrix} \tag{4.21}$$

where l_1, k_1 are arbitrary constants. The system becomes unobservable when:

1. The gyro bias error state vector δb_ω^b is parallel to the specific force vector f^b.

2. The accelerometer bias error state vector δb_a^b is equal to zero.

3. The attitude error state vector $\delta\rho$ is parallel to the specific force vector $f^n = C_b^n f^b$.

If the lever arm vector is considered, the position and velocity error states $(\delta p^n, \delta v^n)$ are unobservable. If the lever arm vector is zero, position and velocity vectors are completely observable.

Therefore, when $ik_1k_{i-1} - k_i \neq 0$, a basis for the null space of (4.18) is:

$$\text{span}\left\{\begin{bmatrix} C_b^n\left(l^b \times f^b\right)_{3\times1} \\ \cdots\cdots \\ 0_{3\times1} \\ \cdots\cdots \\ C_b^n f^b \\ \cdots\cdots \\ 0_{3\times1} \\ \cdots\cdots \\ 0_{3\times1} \end{bmatrix}, \begin{bmatrix} 0_{3\times1} \\ \cdots\cdots \\ C_b^n\left(l^b \times f^b\right)_{3\times1} \\ \cdots\cdots \\ 0_{3\times1} \\ \cdots\cdots \\ 0_{3\times1} \\ \cdots\cdots \\ f_{3\times1}^b \end{bmatrix}\right\} \tag{4.22}$$

Case 2 When $ik_1k_{i-1} - k_i = 0$ or equivalently $k_i = i!k_1^i$, reducing the matrix given in (4.20) will result in the following REF matrix:

$$\begin{bmatrix} -\left[C_b^n f^b \times\right] & C_b^n & 0 \\ 0 & -k_1 C_b^n & -\left[C_b^n f^b \times\right] C_b^n \end{bmatrix} \tag{4.23}$$

Thus, the observability matrix (4.18) has rank 11, the dimension of its null space is 4, and its basis is:

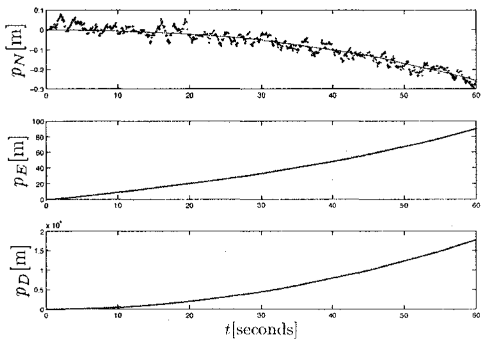
$$
\text{span}\left\{
\begin{bmatrix} -\frac{1}{k_1} C_b^n \left(l^b \times \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \right) \\ C_b^n \left(l^b \times \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \right) \\ -\frac{1}{k_1} C_b^n \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\ -\frac{1}{k_1} \left(f^b \times \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \right) \\ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \end{bmatrix},
\begin{bmatrix} -\frac{1}{k_1} C_b^n \left(l^b \times \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right) \\ C_b^n \left(l^b \times \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right) \\ -\frac{1}{k_1} C_b^n \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \\ -\frac{1}{k_1} \left(f^b \times \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right) \\ \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \end{bmatrix},
\begin{bmatrix} -\frac{1}{k_1} C_b^n \left(l^b \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) \\ C_b^n \left(l^b \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) \\ -\frac{1}{k_1} C_b^n \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ -\frac{1}{k_1} \left(f^b \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) \\ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix},
\begin{bmatrix} 0_{3\times1} \\ C_b^n \left(l^b \times f^b \right) \\ 0_{3\times1} \\ 0_{3\times1} \\ f^b_{3\times1} \end{bmatrix}
\right\}
\tag{4.24}
$$

The Observability conclusions are:

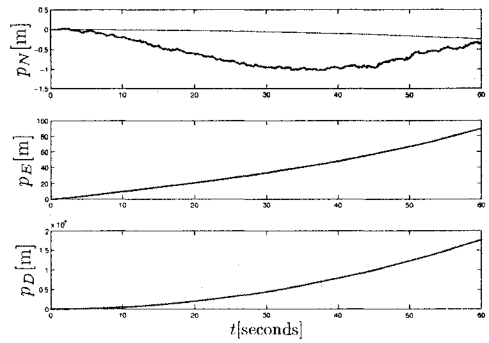
- When the UAV moves with its derivatives in the same direction and satisfies the relation $k_i = i!k_i^i$, the null space has dimension four. The projection of the accelerometer bias error state δb_a^b into the cross product of specific force f^b with gyro bias δb_ω^b is not observable. The projection of the attitude error state $\delta\rho$ into the specific force vector decomposed in the navigation frame coordinates (*i.e.* $f^n = C_b^n f^b$) is not observable.

- When the UAV does not move in the specific fashion described above, the unobservable subspace has dimension two. The projection of the gyro bias δb_ω^b into the specific force vector decomposed in body frame coordinates is not observable. The projection of the attitude error state $\delta\rho$ into the specific force decomposed in navigation frame (i.e. $C_b^n f^b$) is also unobservable.

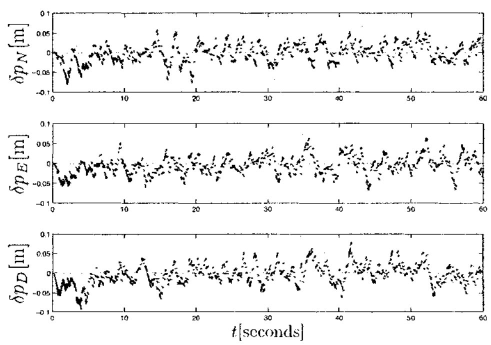
- When lever arm is considered, the projection of the position p^n into the cross product of the lever arm vector and specific force decomposed in navigation frame (i.e. $C_b^n \left(l^b \times f^b\right)$) is unobservable. The same holds for velocity v^n.



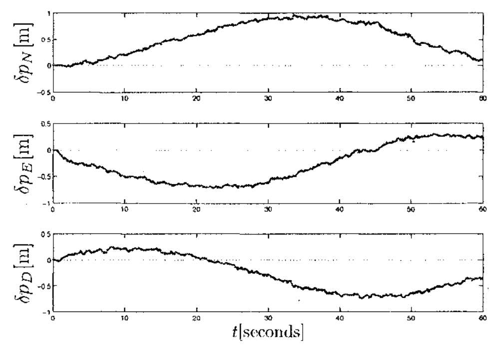
(a) Position states p^n



(b) Position states p^n: lever arm



(c) Position error states δp^n



(d) Position error states δp^n: lever arm

Figure 4.9: Position Estimation

The vehicle has to be accelerating in a certain fashion so that the unobservable subspace dimension is 4. That is we require:

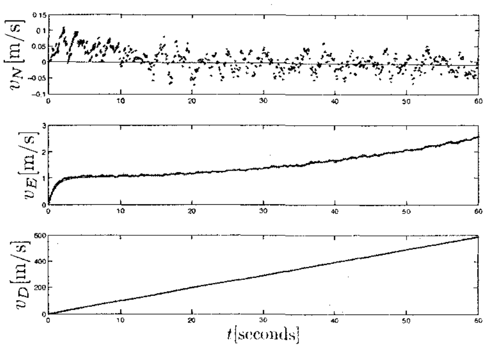
$$\left\{ f^{bT}, \ldots, \overset{(12)}{f^{bT}} \right\} = \left\{ f^{bT}, \ldots, k_{12} f^{bT} \right\} \tag{4.25}$$

$$k_i = i! k_1^i; \quad 1 \leq i \leq 12 \tag{4.26}$$

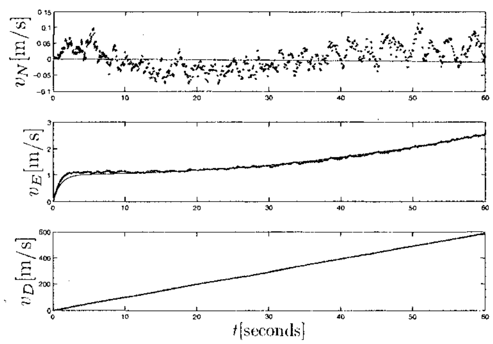
otherwise, the null space dimension is 2. The bases for the null spaces are given in (4.22) and (4.24) respectively.

It is possible to find a trajectory whose time derivatives are in the same direction. However, finding a trajectory which satisfies (4.25) and (4.26) is difficult. Define f^n_{UAV} as the UAV acceleration:

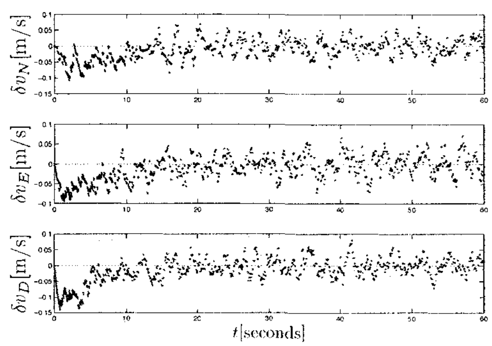
$$f^n = f^n_{UAV} + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \tag{4.27}$$



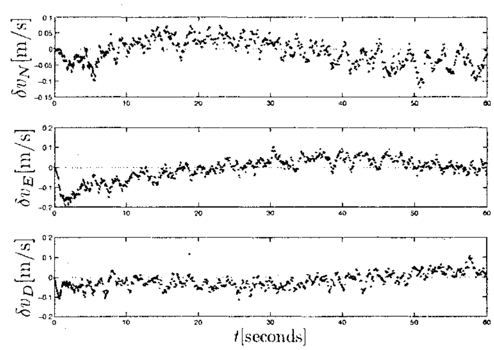
(a) Velocity states v^n



(b) Velocity states v^n: lever arm

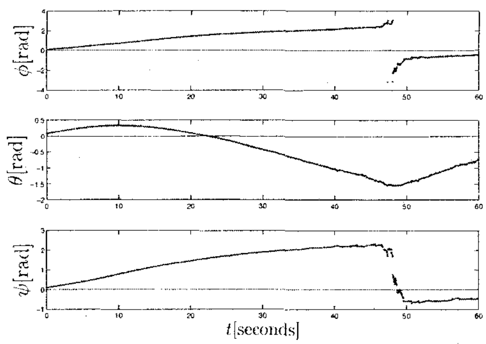


(c) Velocity error states δv^n

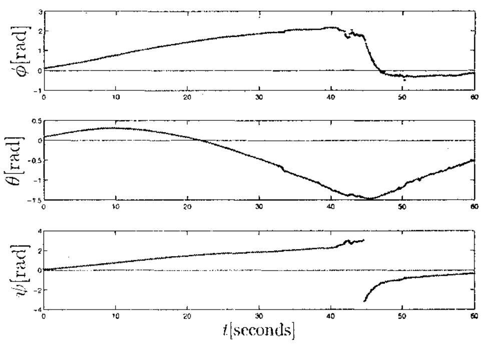


(d) Velocity error states δv^n: lever arm

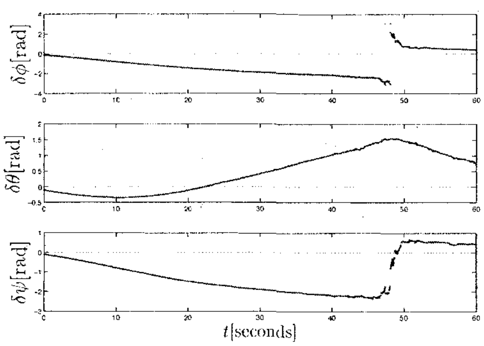
Figure 4.10: Velocity Estimation



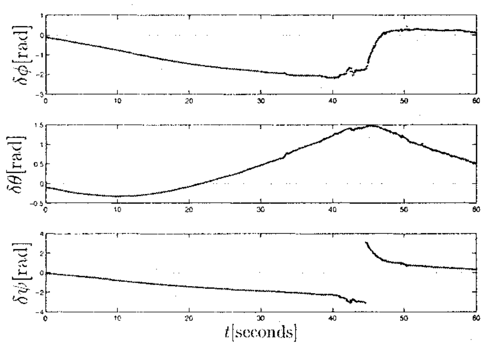
(a) Attitude states ϕ, θ, ψ



(b) Attitude states ϕ, θ, ψ: lever arm

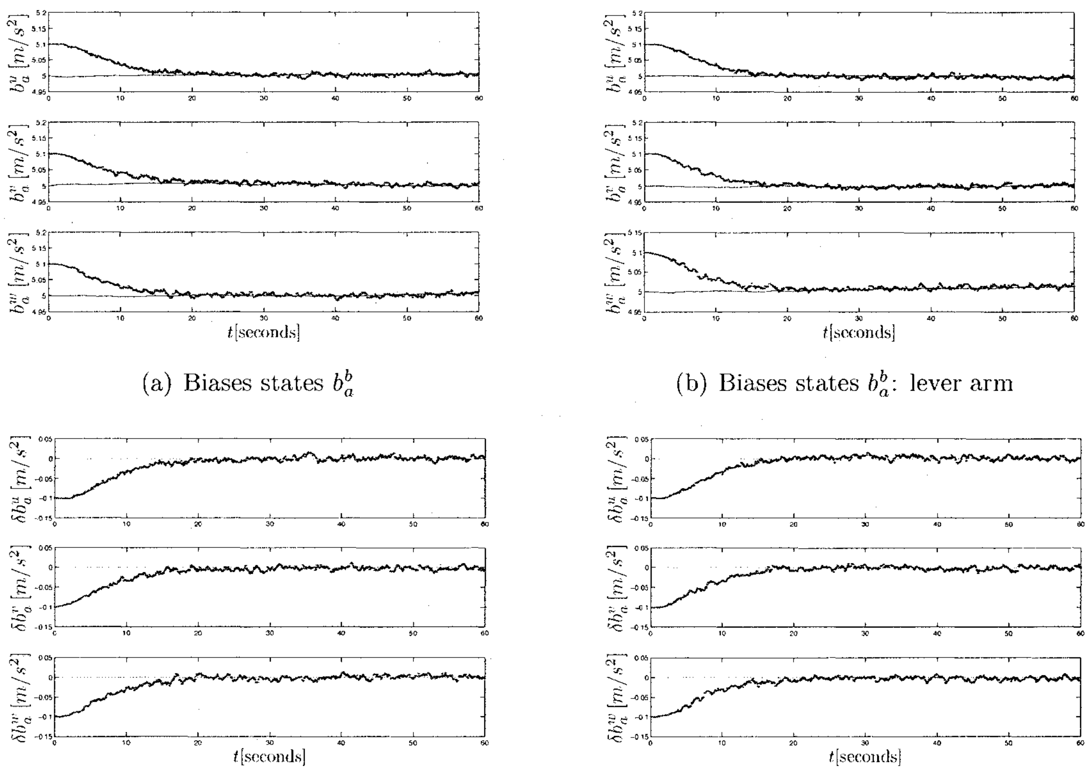


(c) Attitude error states $\delta\rho$



(d) Attitude error states $\delta\rho$: lever arm

Figure 4.11: Attitude Estimation



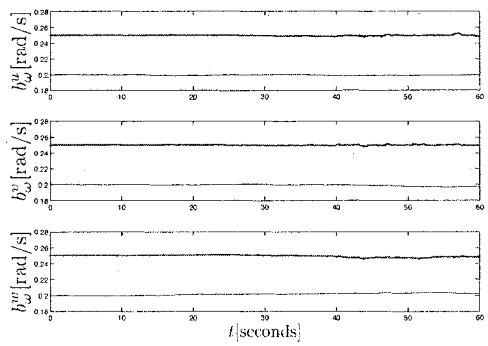
(a) Biases states b_a^b

(b) Biases states b_a^b: lever arm

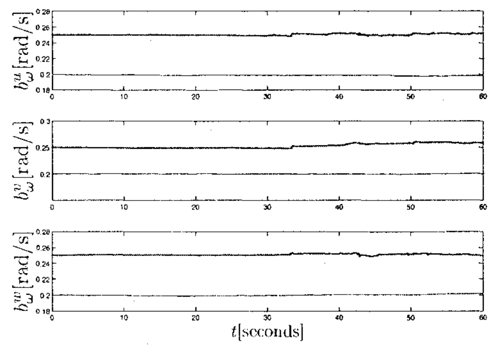
(c) Biases error states δb_a^b

(d) Biases error states δb_a^b: lever arm

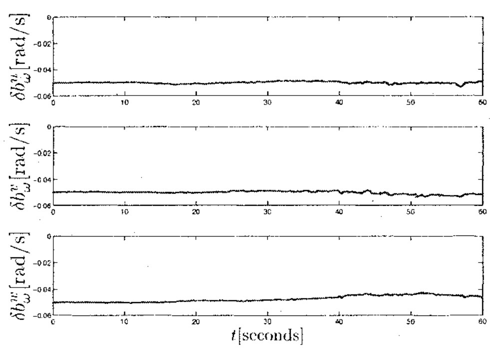
Figure 4.12: Accelerometer Biases Estimation



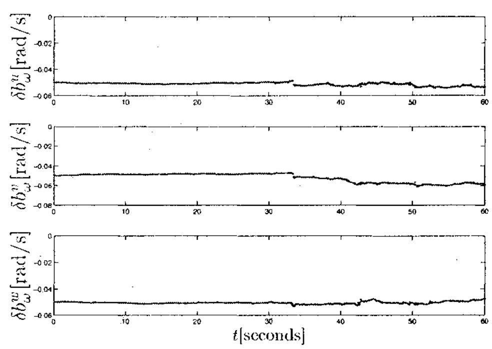
(a) Biases states b^b_ω



(b) Biases states b^b_ω: lever arm



(c) Biases error states δb^b_ω



(d) Biases error states δb^b_ω: lever arm

Figure 4.13: Gyroscope Biases Estimation

Figures 4.9 to 4.13 correspond to an acceleration vector $f^n_{UAV} = \left[0, e^{k_1 t}, g\right]^T$ and $C^n_b = I_{3\times3}$. The specific forces $\left(f^b\right)$ and (f^n) are equal. The time derivatives of f^b satisfy (4.25) but (4.26) does not hold. It is thus expected that the nullity of the observability matrix is two. Problems with attitude and gyro bias determination may arise (see (4.22)) with and without lever arm. Besides attitude and gyroscope biases states, velocity and position states are expected to have convergence problems for nonzero lever arm.

Without lever arm vector, it is evident from Figures 4.13(a) and 4.13(c) that the gyro bias error states δb^b_ω do not converge. Figures 4.11(a) and 4.11(c) confirm the attitude error states $(\delta\rho)$ also do not converge. The remaining error states are observable and they converge to zero.

With the lever arm vector, the accelerometer biases error states (δb^b_a) are the only observable states as seen in Figures 4.12(b) and 4.12(d). The rest of the figures (4.9(b),4.9(d), 4.10(b),4.10(d), 4.11(b),4.11(d), 4.13(b) and 4.13(d)) show non convergence of the error states and agree with (4.22).

4.4.2 Two Time Derivatives of f^b are Linearly Independent

Noting that the set $\left\{f^b, \dot{f}^b, ..., \overset{(12)}{f^b}\right\}$ is spanned by two LI vectors, there is a total of $\binom{13}{2} = 78$ possible sets $\left\{f^b, \dot{f}^b, ..., \overset{(12)}{f^b}\right\}$, where two LI vectors are found.

Let $\overset{(m)}{f^b}, \overset{(q)}{f^b}$ be the vectors that spans such a set. All block submatrices of the observability matrix given in (4.18), excluding the position and velocity error states, can be expressed as a linear combination of the skew symmetric matrix form of vectors $\overset{(m)}{f^b}, \overset{(q)}{f^b}$ as

$$\begin{bmatrix}
-\left(a_{0,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{0,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right) & C_b^n & 0 \\
-\left(a_{1,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{1,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right) & 0 & -\left(a_{0,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{0,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right)C_b^n \\
-\left(a_{2,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{2,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right) & 0 & -2\left(a_{1,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{1,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right)C_b^n \\
\vdots & \vdots & \vdots \\
-\left[C_b^n \overset{(m)}{f^b}\right] & 0 & -m\left(a_{m-1,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{m-1,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right)C_b^n \\
-\left(a_{m+1,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{m+1,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right) & 0 & -(m+1)\left[C_b^n \overset{(m)}{f^b}\right]C_b^n \\
\vdots & \vdots & \vdots \\
-\left[C_b^n \overset{(q)}{f^b}\right] & 0 & -q\left(a_{q-1,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{q-1,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right)C_b^n \\
-\left(a_{q+1,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{q+1,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right) & 0 & -(q+1)\left[C_b^n \overset{(q)}{f^b}\right]C_b^n \\
\vdots & \vdots & \vdots \\
-\left(a_{12,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{12,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right) & 0 & -11\left(a_{11,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{11,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right)C_b^n
\end{bmatrix} \tag{4.28}$$

Assuming that all constants $a_{i,m}, a_{i,q}$ are different from zero we reduce (4.28) into REF:

$$\begin{bmatrix}
-\left(a_{0,m}\left[C_b^n \overset{(m)}{f^b}\times\right]+a_{0,q}\left[C_b^n \overset{(q)}{f^b}\times\right]\right) & C_b^n & 0 \\
a_{0,q}\left[C_b^n \overset{(q)}{f^b}\times\right] & -C_b^n & l^{(m)}_{q,0}\left[C_b^n \overset{(m)}{f^b}\times\right]C_b^n \\
0 & -C_b^n & k^{(m)}_{q,q}\left[C_b^n \overset{(m)}{f^b}\times\right]C_b^n \\
0 & 0 & k^{(m)}_{q,0}\left[C_b^n \overset{(m)}{f^b}\times\right]C_b^n \\
0 & 0 & w^{(q)}_{q,1}\left[C_b^n \overset{(q)}{f^b}\times\right]C_b^n
\end{bmatrix} \tag{4.29}$$

where $k^{(m)}_{q,q}, k^{(m)}_{q,0}, w^{(q)}_{q,1}, l^{(m)}_{q,0}, w^{(m)}_{q,1}$ are constants obtained from row manipulations of the original constants $a_{i,m}, a_{i,q}, [i = 1 \ldots 12]$ and are of no importance in the observability analysis. A further reduction of (4.28) into REF. gives

$$\begin{bmatrix} -a_{0,m}\left[C_b^n \overset{(m)}{f^b}\times\right] & 0 & 0 \\ a_{0,q}\left[C_b^n \overset{(q)}{f^b}\times\right] & 0 & 0 \\ 0 & -C_b^n & 0 \\ 0 & 0 & -k_{q,0}^{(m)}\left[C_b^n \overset{(m)}{f^b}\times\right]C_b^n \\ 0 & 0 & -w_{q,0}^{(m)}\left[C_b^n \overset{(q)}{f^b}\times\right]C_b^n \end{bmatrix} \tag{4.30}$$

With the assumption that all constants in matrix (4.28) are different from zero, we can conclude that the kinematic model is observable as long as:

1. The specific force vector f^b and all its time derivatives can be expressed as a linear combination of the basis $(\overset{(m)}{f^b}, \overset{(q)}{f^b})$.

2. In each of the linear combinations, the two LI vectors found $\overset{(m)}{f^b}, \overset{(q)}{f^b}$ are involved (coefficients different than zero)

Therefore, for the system to be observable, there must be at least four vectors from $\left\{f^b, \dot{f}^b, ..., \overset{(12)}{f^b}\right\}$. Two of the vectors from the former set must be LI while the other two vectors must lay in a linear combination of the basis chosen to make the observability matrix full column rank. It is almost impossible to find a specific force trajectory f^b that has two time derivatives that are LI and the other time derivatives linearly dependent on the former two.

Assume that from the set $\left\{f^b, \dot{f}^b, ..., \overset{(12)}{f^b}\right\}$, f^b and $\dot{f}^b$ are LI and the rest of the time derivatives vanish. Then, the the observability submatrix is

$$\begin{bmatrix} -\left[C_b^n f^b\times\right] & C_b^n & 0 \\ -\left[C_b^n \dot{f}^b\times\right] & 0 & -\left[C_b^n f^b\times\right]C_b^n \\ 0 & 0 & -\left[C_b^n \dot{f}^b\times\right]C_b^n \end{bmatrix} \tag{4.31}$$

The rank of this submatrix is seven and the rank of the observability matrix (4.18) is thirteen. The observability matrix will not be of full column rank and thus the system would be unobservable. A basis for the unobservable subspace is

$$\text{span}\left\{\begin{bmatrix} C_b^n\left(l^b\times f^b\right) \\ C_b^n\left(l^b\times \dot{f}^b\right) \\ C_b^n f^b \\ 0 \\ \dot{f}^b \end{bmatrix}, \begin{bmatrix} C_b^n\left(l^b\times \dot{f}^b\right) \\ 0 \\ C_b^n \dot{f}^b \\ \left(f^b\times \dot{f}^b\right) \\ 0 \end{bmatrix}\right\} \tag{4.32}$$

Observability is improved when three time derivative vectors from the set $\left\{f^b, \dot{f}^b, ..., \overset{(12)}{f^b}\right\}$ are LI.

4.4.3 Three Time Derivatives of f^b are Linearly Independent

It can be shown that O can be made full column rank if at least three vectors of the set $\left\{f^b, \dot{f}^b, \ddot{f}^b, ..., \overset{(12)}{f^b}\right\}$ are LI. Without loss of generality, let $\left\{f^b, \dot{f}^b, \ddot{f}^b\right\}$ be the vectors that span this set. It suffices to check that the rank of

$$\begin{bmatrix} -\left[C_b^n f^b\times\right] & C_b^n & 0 \\ -\left[C_b^n \dot{f}^b\times\right] & 0 & -\left[C_b^n f^b\right] C_b^n \\ -\left[C_b^n \ddot{f}^b\times\right] & 0 & -2\left[C_b^n \dot{f}^b\times\right] C_b^n \end{bmatrix} \tag{4.33}$$

is 9 which implies full column rank. Thus, (4.18) is full column rank. Therefore, when the attitude is kept constant the system can be made observable under the condition that *the specific force and any set of two higher order derivatives are LI.*

4.5 LTV Variable Attitude and Acceleration

The LTV observability matrix given in (4.18) has a more complicated expression when the attitude of the vehicle is changing. A more general form is

$$\begin{bmatrix} I & 0 & -\left[l^n\times\right] & 0 & 0 \\ 0 & I & -\left[\dot{l}^n\times\right] & 0 & -\left[l^n\times\right] C_b^n \\ 0 & 0 & -\left[f^n\times\right]-\left[\ddot{l}^n\times\right] & C_b^n & -2\left[\dot{l}^n\times\right] C_b^n - \left[l^n\times\right] C_b^n\left[\omega_{nb}^b\times\right] \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{4.34}$$

It is a complex task to determine the general conditions for the attitude rate and specific force when the analysis is performed in body frame coordinates. It is then becomes necessary to pick nominal trajectories of the nonlinear system (i.e. equation (3.1)). Some nominal trajectories were taken from reference [2] and are presented in Table 4.1.

Number	f_N	f_E	f_D	ϕ	θ	ψ	$q(O)$
1	$4.9t$	0	9.8	0	0	0	2
2	$4.9t$	$2.45t^2$	9.8	0	0	0	0
3	$-3.84\cos\left(\psi(t)-\frac{\pi}{2}\right)$	$-3.84\sin\left(\psi(t)-\frac{\pi}{2}\right)$	9.8	0	0	$\frac{\pi}{2}+0.8t$	0
4	$-3.84\cos\left(\psi(t)-\frac{\pi}{2}\right)$	$-3.84\sin\left(\psi(t)-\frac{\pi}{2}\right)$	9.8	0	$\frac{\pi t}{18}$	$\frac{\pi}{2}+0.8t$	0
5	$-3.84\cos\left(\psi(t)-\frac{\pi}{2}\right)$	$-3.84\sin\left(\psi(t)-\frac{\pi}{2}\right)$	9.8	$\frac{\pi t}{18}$	0	$\frac{\pi}{2}+0.8t$	0
6	$-4.8\sin\left(\psi(t)-\frac{\pi}{2}\right)$ $-3.84\,(t+1)^2$ $\cos\left(\psi(t)-\frac{\pi}{2}\right)$	$-4.8\cos\left(\psi(t)-\frac{\pi}{2}\right)$ $-3.84\,(t+1)^2$ $\sin\left(\psi(t)-\frac{\pi}{2}\right)$	9.8	0	0	$\frac{\pi}{2}+$ $0.4\left((t+1)^2-1\right)$	0
7	$-4.8\sin\left(\psi(t)-\frac{\pi}{2}\right)$ $-3.84\,(t+1)^2$ $\cos\left(\psi(t)-\frac{\pi}{2}\right)$	$-4.8\cos\left(\psi(t)-\frac{\pi}{2}\right)$ $-3.84\,(t+1)^2$ $\sin\left(\psi(t)-\frac{\pi}{2}\right)$	9.8	0	$\frac{\pi t}{8}$	$\frac{\pi}{2}+$ $0.4\left((t+1)^2-1\right)$	0
8	$-4.8\sin\left(\psi(t)-\frac{\pi}{2}\right)$ $-3.84\,(t+1)^2$ $\cos\left(\psi(t)-\frac{\pi}{2}\right)$	$-4.8\cos\left(\psi(t)-\frac{\pi}{2}\right)$ $-3.84\,(t+1)^2$ $\sin\left(\psi(t)-\frac{\pi}{2}\right)$	9.8	$\frac{\pi t}{8}$	0	$\frac{\pi}{2}+$ $0.4\left((t+1)^2-1\right)$	0

Table 4.1: Observability of Trajectories

Trajectories above give quantities in navigation frame. Hence, the UAV acceleration should be interpreted as f^n and not as f^b. The attitude rate should be taken as the rate of change of the UAV's Euler angles and not as ω_{ib}^b. The attitude rate expressed in body frame $\left(\omega_{nb}^b\right)$ satisfies

$$\begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \sin(\phi)\cos(\theta) \\ 0 & -\sin(\phi) & \cos(\phi)\cos(\theta) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \omega_u \\ \omega_v \\ \omega_w \end{bmatrix} \tag{4.35}$$

The first trajectory has constant attitude. The first two derivatives of specific force f^n are LI and higher order derivatives vanish. The observability analysis reveals that the gyro bias does not converge to its true value using Kalman Filtering because the nullity of O is greater than zero. The null space basis found in equation (4.32) tells us:

1. Gyro bias error δb_ω^b is unobservable in the direction of the specific force f^b.

2. Attitude error $\delta\rho$ is unobservable in two directions which are $C_b^n f^b$ and $C_b^n \dot{f}^b$.

3. Accelerometer bias error is unobservable in the direction of the cross product of specific force and its first order time derivative $f^b \times \dot{f}^b$.

4. Without lever arm, position and velocity error states are completely observable. With lever arm position error is unobservable in the directions of $C_b^n\left(l^b \times f^b\right)$ and $C_b^n\left(l^b \times \dot{f}^b\right)$, velocity error is unobservable in the direction of $C_b^n\left(l^b \times \dot{f}^b\right)$.

In all other trajectories, it can verified that at least three time derivatives of the specific force in navigation frame f^n are LI, rendering the system completely observable. A Kalman Filter should converge for trajectories 2 to 8.

In order to obtain time varying accelerations (variable f^n) in practice, it is easier to rotate the UAV rather than increasing/decreasing the acceleration while vehicle moves along a straight line.

Trajectories 3 to 5 are the easiest to perform in a real flight because there is constant angular velocity and the UAV is moving along a circle. Furthermore a UAV flying at constant speed around circles will give time varying accelerations and all fifteen states to be estimated.

Trajectories 1 and 2 are difficult to perform because they involve an increase in the acceleration which makes the vehicle accelerate faster besides moving faster in each time instant. Trajectories 6 to 8 are problematic due to reasons similar to those explained for the first and second trajectories. Furthermore, the angular velocity ω_{nb}^b is not constant and increases with time.

Chapter 5

Magnetometer/GPS/INS Attitude Determination

The observability analysis performed previously tells us that a UAV must maneuver to estimate its state correctly. The states that are affected the most are the attitude states (roll, pitch, yaw). A third sensor is needed in order to estimate the system's states for any vehicle trajectory. This chapter illustrates how a magnetometer is used to eliminate the observability problems of the GPS-aided approach.

5.1 EKF With Raw Measurements

Estimates of pitch θ and roll ϕ angles can be obtained with accelerometer sensors [6, Sec. 9]. The specific force measured by the accelerometers is equal to gravity when the UAV is not accelerating. This situation happens when the UAV is on the ground, hovering, or going at a constant velocity, in other words if the UAV is stationary. Furthermore, letting the navigation frame of the UAV to be the local geographic plane, the specific force vector measured by strap-down sensors should be the gravity vector, which is well defined in this frame as $[0, 0, -g]^T \frac{\text{m}}{\text{s}^2}$ and is rotated by the current attitude of the vehicle to obtain the gravity vector in body frame coordinates.

Reference [4, Ch. 6.8.1.1] provides an explanation on how to estimate θ and ϕ from accelerometer measurements. Knowing θ and ϕ, measurement of the earth's magnetic field suffices to determine the heading angle or yaw (ψ).

We have

$$C_n^b = \begin{bmatrix} \cos(\psi)\cos(\theta) & \sin(\psi)\cos(\theta) & -\sin(\theta) \\ -\sin(\psi)\cos(\phi)+\cos(\psi)\sin(\theta)\sin(\phi) & \cos(\psi)\cos(\phi)+\sin(\psi)\sin(\theta)\sin(\phi) & \cos(\theta)\sin(\phi) \\ \sin(\psi)\sin(\phi)+\cos(\psi)\sin(\theta)\cos(\phi) & -\cos(\psi)\sin(\phi)+\sin(\psi)\sin(\theta)\cos(\phi) & \cos(\theta)\cos(\phi) \end{bmatrix} \tag{5.1}$$

and so

$$C_b^n = \begin{bmatrix} \cos(\psi)\cos(\theta) & -\sin(\psi)\cos(\phi)+\cos(\psi)\sin(\theta)\sin(\phi) & \sin(\psi)\sin(\phi)+\cos(\psi)\sin(\theta)\cos(\phi) \\ \sin(\psi)\cos(\theta) & \cos(\psi)\cos(\phi)+\sin(\psi)\sin(\theta)\sin(\phi) & -\cos(\psi)\sin(\phi)+\sin(\psi)\sin(\theta)\cos(\phi) \\ -\sin(\theta) & \cos(\theta)\sin(\phi) & \cos(\theta)\cos(\phi) \end{bmatrix} \tag{5.2}$$

Accelerometers measure specific force in body frame f^b. In case the UAV is accelerating, the specific force measurement will therefore be as in (3.12):

$$f^b = \left(\frac{d^2p^b}{dt^2}\right)_i + g^b \tag{5.3}$$

It is necessary to subtract the vehicle's acceleration vector $f^b_{UAV} = \left(\frac{d^2p^b}{dt^2}\right)_i$ from the specific force vector by means of external sensors like differentiating a GPS velocity measurement.

After proper compensation for UAV's acceleration or if the UAV is stationary, the IMU's accelerometers measure the gravity vector decomposed in body frame as follows:

$$\tilde{f}^b = \begin{bmatrix} \tilde{f}_u \\ \tilde{f}_v \\ \tilde{f}_w \end{bmatrix} = -C_n^b \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} g\sin(\theta) \\ -g\cos(\theta)\sin(\phi) \\ -g\cos(\theta)\cos(\phi) \end{bmatrix} \tag{5.4}$$

Thus, ϕ and θ are obtained from

$$\begin{bmatrix} \phi \\ \theta \end{bmatrix} = \begin{bmatrix} \arctan 2\left(-\tilde{f}_v, -\tilde{f}_w\right) \\ \arctan 2\left(\tilde{f}_u, \sqrt{\tilde{f}_v^2 + \tilde{f}_w^2}\right) \end{bmatrix} \tag{5.5}$$

The Earth's magnetic field is modeled as a magnetic dipole. Magnetic field lines converge to magnetic north and diverge from magnetic south.

Define a magnetic coordinate frame as the rotation of the fixed tangent plane frame about its down axis by a declination angle λ. Therefore the fixed tangent plane and the magnetic frames differ in their x and y axes. The tangent plane frame north aims towards the true north of earth whereas the magnetic frame north aims to the magnetic north. Figure 5.1 shows a magnetic field vector measurement in magnetic frame coordinates h^m. Assume the symbols x, y, z are the axes of the magnetic frame. h_h^m represents the projection of h into the fixed tangent plane frame. h_x^m, h_y^m and h_z^m represent respectively the forward, rightward and downward magnetic components of the sensor measurement. The symbol α represents the angle between the magnetic frame's forward axis x and h_h^m. The symbol λ represent the declination angle and the symbol δ the inclination angle of the magnetic field.

The angle made between earth's true north and earth's magnetic north axes is known as declination and is approximately $16.35^0 E$ in Edmonton. The declination angle is computed based on the earth's magnetic field model.

Inclination δ and declination λ angles have to be taken into account for an accurate computation of the heading or yaw ψ with respect to the true north. Following the right hand rule, the rotation measured by the gyroscopes is positive when counterclockwise. Therefore, in order to keep this convention with the compass sensor, a positive angle α will have h_h^m to the east of x. Conversely a negative α will have h_h^m to the west of x.

Magnetometers measure the magnetic field intensity in its orthogonal axes which are assumed to be parallel to the body frame axes. The horizontal components of the magnetic field decomposed in geographic frame determine the heading direction with respect to the magnetic north. To determine the yaw ψ with respect to the true north, subtract α from 16.35^0 as follows [6]:

$$\psi = \frac{16.35°\pi}{180°} - \arctan 2\left(\frac{h_y^m}{h_x^m}\right) \tag{5.6}$$

To determine the magnitude of the magnetic field in the direction perpendicular to the gravity vector, it is necessary to combine the roll ϕ and pitch θ estimates given by the accelerometer measurements (equation (5.5)) with the earth's magnetic field given by the magnetometer measurements. Reference [6, Sec. 9] shows the computation of the horizontal earth field components perpendicular to the gravity vector. These components are computed from the measured body frame quantities using the following formulas:

$$h_x^m = h_u^b \cos(\theta) - h_v^b \sin(\theta)\sin(\phi) - h_w^b \sin(\theta)\cos(\phi) \tag{5.7}$$

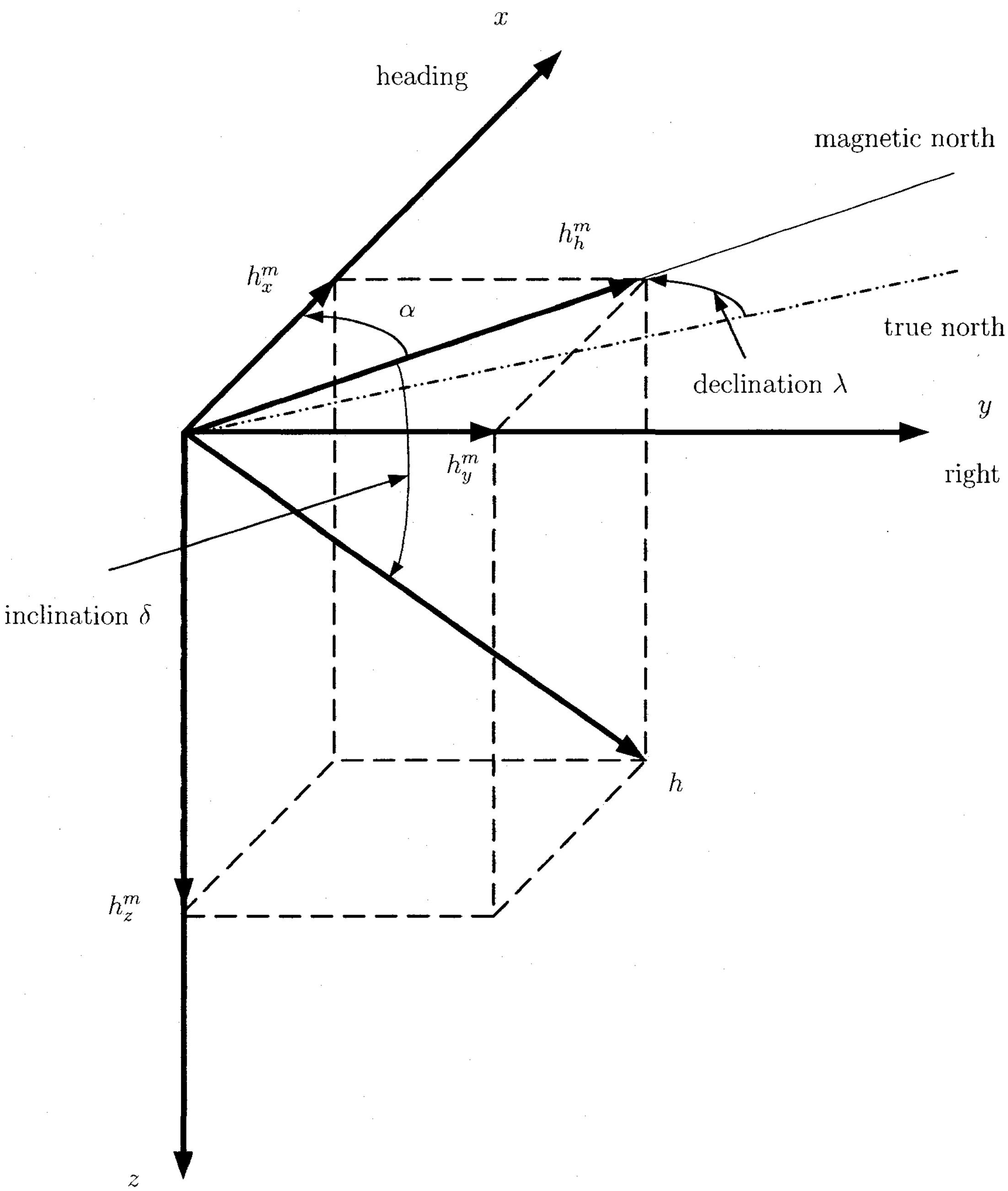


Figure 5.1: Earth Magnetic Field

$$h_y^m = h_v^b \cos(\phi) + h_w^b \sin(\phi) \tag{5.8}$$

The attitude of the UAV is fully determined without the integration of the gyro rates. This method of determination works if UAV is not accelerating. If vehicle is maneuvering, the incorporation of gyro rate integration is needed. If the UAV is not accelerating, the INS sensor can combine the accelerometer and magnetometer readings and compute the attitude of the vehicle using equations (5.5) to (5.8).

When the navigation system is able to measure the position and attitude, the linear error states of the kinematic model, including gyro and accelerometer biases, become fully observable. As opposed to gyroscopes and accelerometers, magnetometer readings are not integrated over time. Therefore, the magnetometer readings can be used as an aiding source in the attitude determination along with GPS because of their bounded error characteristics.

This approach is not reliable because of the error sources discussed thoroughly in references [7] and [8]. Example of such sources are:

- Sensor offset is a constant bias at the output
- Scale factor miss match is another source of error where the triaxial measurements are slightly coupled.
- Non orthogonality errors occur when the axes of magnetometers are not perfectly orthogonal.
- Sensor tilts occur when the magnetometers are not perfectly leveled with the local geographic coordinate frame.

An EKF will improve in a covariance sense the attitude determination. The following section will describe one EKF approach.

5.1.1 Kalman Filtering

The fifteen navigation states can be determined using the IMU's accelerometer and magnetometer measurements and the GPS position measurements.

The measurement update phase is done at the rate of the GPS system. The algorithm which is based on the one written in reference [4, Ch. 4.7.5.2] is presented:

1. Determine the input noise covariance matrix Q.

2. Define the state $\hat{x}$ as:

$$\hat{x} = \begin{bmatrix} \hat{p}^n \\ \hat{v}^n \\ \hat{C}_b^n \\ \hat{b}_a^b \\ \hat{b}_\omega^b \end{bmatrix} \tag{5.9}$$

3. Determine the initial conditions of the state $\hat{x}(0)$. They could assume any value. One alternative is $\hat{p}^n(0) = 0$, $\hat{v}^n(0) = 0$, $\hat{C}_b^n(0) = I_{3\times3}$, $\hat{b}_a^b(0) = 0$, $\hat{b}_\omega^b(0) = 0$.

4. Determine the initial condition of the predicted covariance matrix $P^-(0)$. One alternative is $P^-(0) = 0_{15\times15}$.

5. The UAV attitude is determined by the Microstrain sensor as follows:

$$\begin{bmatrix} \tilde{\phi} \\ \tilde{\theta} \end{bmatrix} = \begin{bmatrix} \arctan 2\left(-\tilde{f}_v, -\tilde{f}_w\right) \\ \arctan 2\left(\tilde{f}_u, \sqrt{\tilde{f}_v^2 + \tilde{f}_w^2}\right) \end{bmatrix} \tag{5.10}$$

$$\tilde{h}_x^m = \tilde{h}_u^b \cos\left(\tilde{\theta}\right) - \tilde{h}_v^b \sin\left(\tilde{\theta}\right) \sin\left(\tilde{\phi}\right) - \tilde{h}_w^b \sin\left(\tilde{\theta}\right) \cos\left(\tilde{\phi}\right) \tag{5.11}$$

$$\tilde{h}_y^m = \tilde{h}_v^b \cos\left(\tilde{\phi}\right) + h_w^b \sin\left(\tilde{\phi}\right) \tag{5.12}$$

$$\tilde{\psi} = \frac{16.35^0 \pi}{180} - \arctan 2\left(\frac{\tilde{h}_y^m}{\tilde{h}_x^m}\right) \tag{5.13}$$

6. Every time there is a GPS measurement, define the measurement equation as:

$$\tilde{y} = \begin{pmatrix} \tilde{p}_{GPS}^n \\ \tilde{\phi} \\ \tilde{\theta} \\ \tilde{\psi} \end{pmatrix} \tag{5.14}$$

7. Every time there is a GPS measurement, define the estimated measurement equation as [1]:

$$\hat{y} = \begin{pmatrix} \hat{p}^n + \hat{C}_b^n l^b \\ \hat{\phi} \\ \hat{\theta} \\ \hat{\psi} \end{pmatrix} \tag{5.15}$$

8. Every time there is a GPS measurement, define the error as $\delta y = \tilde{y} - \hat{y}$.

9. Every time there is a GPS measurement, define the Output matrix as

$$H = \begin{bmatrix} I & 0 & -\left[\hat{C}_b^n l^b \times\right] & 0 & 0 \\ 0 & 0 & I & 0 & 0 \end{bmatrix} \tag{5.16}$$

10. Every time there is a GPS measurement, update the Kalman gain as

$$K = P^- H^T \left[R + H P^- H^T\right]^{-1} \tag{5.17}$$

11. Every time there is a GPS measurement, correct the predicted state as $\hat{x} = \hat{x}^- + K\delta y$.

12. Every time there is a GPS measurement, correct the predicted state covariance matrix as $P = (I - KH)\, P^-$.

13. Every time there is a GPS measurement, define the state matrix F as:

$$F = \begin{bmatrix} 0 & I & 0 & 0 & 0 \\ 0 & -2\Omega_{ie}^n & -\left[\hat{C}_b^n \hat{f}^b \times\right] & \hat{C}_b^n & 0 \\ 0 & 0 & -\Omega_{ie}^n & 0 & \hat{C}_b^n \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.18}$$

14. Every time there is a GPS measurement, define the input noise matrix B as:

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \hat{C}_b^n & 0 & 0 & 0 \\ 0 & \hat{C}_b^n & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \tag{5.19}$$

[1] Appendix A explains how to estimate roll, pitch and yaw (ϕ, θ, ψ) from direction cosine matrix C_b^n

15. Every time there is a GPS measurement, define matrix A as:

$$A = \begin{bmatrix} -F & BQB^T \\ 0 & F^T \end{bmatrix} \tag{5.20}$$

16. Every time there is a GPS measurement, compute $\Upsilon = e^{(A\Delta T_{GPS})}$. Then find Φ and Q_d.

17. Every time there is a GPS measurement, propagate the predicted covariance matrix as $P^- = \Phi P \Phi^T + Q_d$.

18. Every time there is an IMU measurement, propagate the state using any numerical integration method of the following first order differential equation:

$$\begin{pmatrix} \dot{\hat{p}}^n \\ \dot{\hat{v}}^n \\ \dot{\hat{C}}_b^n \\ \dot{\hat{b}}_a^b \\ \dot{\hat{b}}_\omega^b \end{pmatrix} = \begin{pmatrix} \hat{v}^n \\ \hat{C}_b^n \hat{f}^b + g^n - 2\Omega_{ie}^n \hat{v}^n \\ \hat{C}_b^n \hat{\Omega}_{nb}^b \\ 0 \\ 0 \end{pmatrix} \tag{5.21}$$

19. Go back to step 5.

5.1.2 Final Remarks

A navigation system must provide reliable position, velocity and orientation (attitude) data. However position and velocity estimation can be separated from attitude determination with GPS sensors. Attitude is a more complex problem to solve.

Existing work reduce the order of the navigation equations, use more measurements than what are available at the ANCL Lab or deal with the attitude as a separate problem. References [9] and [10] explain how to solve the attitude determination problem based on: (1) Magnetic field and specific force measurements and (2) Multiple GPS antenna. References [11] and [12] solve the same fifteen state navigation estimation problem described in this thesis based on: (1) Magnetometer and GPS measurements and (2) Multiple GPS antenna, with a tightly coupled approach. Reference [13] analyzes a reduced order navigation problem excluding position states.

This thesis uses magnetometers, accelerometers and GPS measurements in order to aid a fifteen state INS in a loosely coupled approach. For the scheme of attitude determination described above, it is assumed that all Euler angles are measured. Equations (5.10) to (5.13) are used to determine the measured values of the euler angles (ϕ, θ, ψ). The approach described in 5.1.1 will work only if the UAV is stationary. In case the UAV is accelerating or rotating, the specific force vector f^b is not gravity and attitude can not be determined.

Furthermore the latter approach assumes that the measurements of specific force and magnetic field are unbiased and corrupted preferably by white noise or any other source of stable noise in the covariance sense (the covariance of random walk noise is unbounded for large time). The IMU sensor being used by the ANCL Lab has DC offset bias in all its sensors. This is the biggest source of error because the integration over time will result in an unbounded solution.

5.2 Magnetometer/Accelerometer/GPS EKF

5.2.1 Attitude Linear Error State

The attitude kinematics are

$$\dot{C}_b^n = C_b^n \Omega_{nb}^b \tag{5.22}$$

The estimated version for attitude kinematics are

$$\dot{\hat{C}}_b^n = \hat{C}_b^n \hat{\Omega}_{nb}^b \tag{5.23}$$

The rate of change of the attitude is the rotation vector of the body frame with respect to the fixed tangent plane frame Ω_{nb}^b. This vector is partially measured by the gyroscopes of the IMU. However, it is necessary to compensate the earth rotation with respect to the inertial frame Ω_{ie}^n since the measurements made by the sensors are inertial. This vector decomposed in body frame is

$$\Omega_{nb}^b = \Omega_{ib}^b - C_n^b \Omega_{ie}^n \tag{5.24}$$

The estimated version of the rotation vector mentioned above, assuming the earth rotation vector is known is

$$\hat{\Omega}^b_{nb} = \hat{\Omega}^b_{ib} - \hat{C}^b_n \Omega^n_{ie} \tag{5.25}$$

Recalling that $x - \hat{x} = \delta x$, $C^b_n = C^{n^T}_b$ and $\hat{C}^n_b = (I - [\delta\rho\times])\, C^n_b$, where $[\delta\rho\times]$ is the skew symmetric form of error in angle deviation of the computed body frame coordinate system with the actual body frame coordinate system. A linear model for attitude rate of change can be found in terms of gyro sensor readings. From (5.23):

$$(I - [\delta\rho\times])\, C^n_b \Omega^b_{nb} - [\delta\dot{\rho}\times]\, C^n_b = (I - [\delta\rho\times])\, C^n_b \hat{\Omega}^b_{nb} \tag{5.26}$$

Solving for $[\delta\dot{\rho}\times]$ and neglecting small terms involving products of differential quantities (i.e. $[\delta\rho\times]\, C^n_b \delta\Omega^b_{nb}$), a first order approximation of attitude error between computed and real body frames is

$$[\delta\dot{\rho}\times] = C^n_b \delta\Omega^b_{nb} C^b_n \tag{5.27}$$

To determine an error model for the rotation rate $\delta\Omega^b_{nb}$, it suffices to subtract (5.25) from (5.24) and notice that $\hat{C}^b_n = C^b_n (I + [\delta\rho\times])$:

$$\delta\Omega^b_{nb} = \delta\Omega^b_{ib} - C^b_n \Omega^n_{ie} [\delta\rho\times] \tag{5.28}$$

The model for gyro rate Ω^b_{ib} is:

$$\Omega^b_{ib} = \tilde{\Omega}^b_{ib} + \left[b^b_\omega\times\right] + [\nu_\omega\times] \tag{5.29}$$

where the tilde sign indicates sensor measurement and ν indicates sensor noise. The estimated version of (5.29) is

$$\hat{\Omega}^b_{ib} = \tilde{\Omega}^b_{ib} + \left[\hat{b}^b_\omega\times\right] \tag{5.30}$$

Subtracting (5.30) from (5.29) and replacing this result into (5.28) we obtain:

$$\delta\Omega^b_{nb} = -C^b_n \Omega^n_{ie} + \left[\delta b^b_\omega\times\right] + [\nu_\omega\times] \tag{5.31}$$

Finally, (5.27) expressed in vector form is:

$$\delta\rho = -\omega^n_{ie} \times \delta\rho + C^n_b \delta b^b_\omega + C^n_b \nu_\omega \tag{5.32}$$

Sensor biases are typically modeled as Gauss-Markov processes, however for the sake of simplicity in the observability analysis, they will be modeled as random walk processes. Letting $\left[\delta\rho^T, \delta b_a^{b^T}, \delta b_\omega^{b^T}\right]^T$ be the state variables, the linear error state equation is

$$\begin{bmatrix} \delta\dot{\rho} \\ \delta\dot{b}_a^b \\ \delta\dot{b}_\omega^b \end{bmatrix} = \begin{bmatrix} -\Omega_{ie}^n & 0 & C_b^n \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta\rho \\ \delta b_a^b \\ \delta b_\omega^b \end{bmatrix} + \begin{bmatrix} C_b^n & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} \nu_\omega \\ \nu_{b_a} \\ \nu_{b_\omega} \end{bmatrix} \tag{5.33}$$

5.2.2 EKF Implementation with Lever Arm

The output will include magnetic and acceleration measurements. As opposed to the acceleration measurement, the magnetic measurement is not integrated over time and bounded biased solutions of attitude can be obtained. The accuracy depends on the low frequency components of noise from the magnetometer sensor, which will not be corrected online, and they will be assumed equal to zero or in other words, our designed EKF will not make any effort in trying to estimate the magnetometer noise.

Data on the real earth magnetic field at any position (latitude, longitude, altitude) can be found at several references. The National Oceanic and Atmospheric Administration (NOAA) and the International Association of Geomagnetism and Aeronomy (IAGA) are examples of online databases that provide the earth magnetic field vector decomposed in geographic frame at any location on the planet with maximum resolution of arcseconds or 30m of distance. The earth's magnetic field vector does not vary significantly within a few hundred of kilometers. Thus, this vector is assumed to be known and constant near the location where the UAV is flying.

In order to find the earth's magnetic field, the position of the base station GPS antenna needs to be determined. Once the geodetic coordinates are obtained (latitude, longitude and height), the magnetic field can be determined by applying the gradient of a scalar potential V represented by a truncated series expansion[2]. Given the latitude ($53^0\,31'\,37''N$) longitude ($113^0\,31'\,49''W$) and height ($707m$) parameters, the NOAA online calculator will compute the

[2]http://www.ngdc.noaa.gov/IAGA/ vmod/igrf.html

magnetic field vector. Scalar potential V computation and GPS positioning are explained in appendix C and D of this thesis.

There will be three output vectors: the GPS position (p^n_{GPS}), the gravitational field vector (g^n), and the earth magnetic field vector at the point of origin of the tangent plane of interest ($h^n(\phi, \lambda, h)$). Keep in mind that the GPS antenna is shifted from the IMU sensor by a vector called the lever arm l^b, which should be known and specified in body frame coordinates.

$$y = \begin{pmatrix} \tilde{p}^n_{GPS} \\ g^n \\ h^n \end{pmatrix} \tag{5.34}$$

From (5.34), the only measured quantity is the position. The gravity and magnetic field vectors (g^n, h^n respectively) are known quantities and equal to:

$$g^n = \begin{bmatrix} 0 \\ 0 \\ -9.8 \end{bmatrix} \frac{\mathrm{m}^2}{\mathrm{s}} \tag{5.35}$$

$$h^n = \begin{bmatrix} 0.139 \\ 0.041 \\ 0.562 \end{bmatrix} \text{Gauss} \tag{5.36}$$

When the UAV accelerates, it is necessary to compensate for these motions since the specific force vector f^b measured by the accelerometers will no longer be equal to the gravity vector g^n.

First Order Accelerometer Output Equation

Letting f^n_{UAV} be the UAV acceleration vector decomposed in navigation frame, then the gravity vector g^n is equivalent to

$$g^n = C^n_b f^b - f^n_{UAV} \tag{5.37}$$

The UAV acceleration vector can be obtained indirectly from GPS sensors. The GPS sensor position p^n_{GPS} is

$$p^n_{GPS} = p^n + C^n_b l^b \tag{5.38}$$

The first order time derivative of equation (5.38) is the GPS sensor velocity v^n_{GPS}

$$v^n_{GPS} = \dot{p}^n + C^n_b \left(\omega^b_{nb} \times l^b\right) \tag{5.39}$$

The first order time derivative of equation (5.39) is the GPS sensor acceleration $\dot{v}^n_{GPS}$

$$\dot{v}^n_{GPS} = \ddot{p}^n + C^n_b \left(\omega^b_{nb} \times \left(\omega^b_{nb} \times l^b\right)\right) + C^n_b \left(\dot{\omega}^b_{nb} \times l^b\right) \tag{5.40}$$

p^n is the UAV position, then $\ddot{p}^n$ is the UAV acceleration. Solving (5.40) for $\ddot{p}^n$, the UAV acceleration equation is

$$f^n_{UAV} = \dot{v}^n_{GPS} - C^n_b \left(\omega^b_{nb} \times \left(\omega^b_{nb} \times l^b\right)\right) - C^n_b \left(\dot{\omega}^b_{nb} \times l^b\right) \tag{5.41}$$

The estimated version of the UAV acceleration vector f^n_{UAV} is

$$\hat{f}^n_{UAV} = \dot{v}^n_{GPS} - \hat{C}^n_b \left(\omega^b_{nb} \times \left(\omega^b_{nb} \times l^b\right)\right) - \hat{C}^n_b \left(\dot{\omega}^b_{nb} \times l^b\right) \tag{5.42}$$

The specific force vector as read by the accelerometer sensors ($\tilde{f}^b$) is corrupted by a DC offset, random walk noise and white noise. Only DC offset can be compensated for off-line. The random walk noise corruption is neglected as long as its variance is much smaller than the white noise variance during the time the UAV is operating $\left(i.e.\ \sigma^2_{\nu_{b^b_a}} t \ll \sigma^2_{\nu^b_a}\right)$. The sensor measurement after DC offset compensation can be approximated by the actual specific force corrupted by white noise ($i.e.\ \tilde{f}^b = f^b - \nu^b_a$). This same argument applies for the gyroscope sensor measurements ($\tilde{\omega}^b_{ib}$) and magnetometer sensor measurements ($\tilde{h}^b$).

The estimated gravity vector is:

$$\hat{g}^n = \hat{C}^n_b \left(\tilde{g}^b + \tilde{f}^b_{UAV}\right) - \underbrace{\left(\dot{v}^n_{GPS} - \hat{C}^n_b \left(\omega^b_{nb} \times \left(\omega^b_{nb} \times l^b\right)\right) - \hat{C}^n_b \left(\dot{\omega}^b_{nb} \times l^b\right)\right)}_{\hat{f}^n_{UAV}} \tag{5.43}$$

Further developing (5.43) results in

$$\hat{g}^n = \left(I - [\delta\rho\times]\right) C^n_b \tilde{g}^b + \left(I - [\delta\rho\times]\right) \underbrace{C^n_b \tilde{f}^b_{UAV}}_{\tilde{f}^n_{UAV}} - \hat{f}^n_{UAV} \tag{5.44}$$

Equation (5.42) can be expressed as

$$\hat{f}^n_{UAV} = \dot{v}^n_{GPS} - \left(I - [\delta\rho\times]\right) C^n_b \left(\omega^b_{nb} \times \left(\omega^b_{nb} \times l^b\right)\right) - \left(I - [\delta\rho\times]\right) C^n_b \left(\dot{\omega}^b_{nb} \times l^b\right) \tag{5.45}$$

Reducing equation (5.44) gives

$$\hat{g}^n = (I - [\delta\rho\times])\, C_b^n \tilde{g}^b - [\delta\rho\times]\, \dot{\tilde{v}}^n_{GPS} = (I - [\delta\rho\times])\, \tilde{g}^n - [\delta\rho\times]\, \dot{\tilde{v}}^n_{GPS} \qquad (5.46)$$

Finally the linear residual part due to accelerometers is

$$\delta g^n = g^n - \hat{g}^n = -\left(g^n + \dot{v}^n_{GPS}\right) \times \delta\rho - \nu_a - \frac{d\nu_{v^n_{GPS}}}{dt} \qquad (5.47)$$

And the estimated output gravity vector is

$$\hat{g}^n = \hat{C}_b^n \tilde{f}^b - \hat{f}^n_{UAV} \qquad (5.48)$$

First Order Position Output Equation

The UAV center of mass position is indirectly measured by the GPS sensor. Recalling equation (5.38):

$$\tilde{p}^n_{GPS} = \tilde{p}^n + C_b^n l^b = p^n + C_b^n l^b + \nu_p \qquad (5.49)$$

The estimated version of (5.49) is

$$\hat{p}^n_{GPS} = \hat{p}^n + \hat{C}_b^n l^b \qquad (5.50)$$

Which can be expressed as

$$\hat{p}^n_{GPS} = \hat{p}^n + (I - [\delta\rho\times])\, C_b^n l^b \qquad (5.51)$$

Finally subtracting (5.51) from (5.49) results in the residual output due to the GPS sensor:

$$\delta p^n_{GPS} = \delta p^n - \left(C_b^n l^b\right) \times \delta\rho + \nu_p \qquad (5.52)$$

The estimated output position vector is:

$$\hat{p}^n_{GPS} = \hat{p}^n + \hat{C}_b^n l^b \qquad (5.53)$$

First Order Magnetometer Output Equation

The earth magnetic field model as stated before is known. The magnetometer measures the earth magnetic field vector decomposed in body frame coordinates. As with the accelerometers, these measurements are corrupted by DC offset bias, white noise and other sources of noise such as random walk, flickering noise, quantization noise, rate ramp noise and Gauss Markov process noise.

DC offset bias can be easily compensated for off line. However, compensation for noise requires an augmentation of the state space. For example, the state space will have dimension eighteen when the magnetic measurements are assumed to be corrupted by random walk noise. All states will not be estimated correctly because the system will become unobservable, in particular the magnetic measurement biases.

It is therefore required to assume that the magnetometer sensor measurements are corrupted by white noise plus a Gauss-Markov noise process. The Earth magnetic field vector is:

$$h^n = C_b^n h^b \tag{5.54}$$

The estimated earth magnetic field vector is

$$\hat{h}^n = \hat{C}_b^n \tilde{h}^b = (I - [\delta\rho\times])\, C_b^n h^b - \underbrace{(I - [\delta\rho\times])\, C_b^n \nu_m^b}_{\nu_m^n} \tag{5.55}$$

The residual magnetic field output vector is

$$\delta h^n = h^n - \hat{h}^n = [\delta\rho\times]\, C_b^n h^b - \nu_m^n = -h^n \times \delta\rho - \nu_m^n \tag{5.56}$$

The estimated output magnetic field vector is

$$\hat{h}^n = \hat{C}_b^n \tilde{h}^b \tag{5.57}$$

State Space Implementation

The complete linear error state model is

$$\begin{bmatrix} \delta\dot{p}^n \\ \delta\dot{v}^n \\ \delta\dot{\rho} \\ \delta\dot{b}_a^b \\ \delta\dot{b}_\omega^b \end{bmatrix} = \begin{bmatrix} 0 & I & 0 & 0 & 0 \\ 0 & -2\Omega_{ie}^n & -\left[C_b^n f^b \times\right] & C_b^n & 0 \\ 0 & 0 & -\Omega_{ie}^n & 0 & C_b^n \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta\rho \\ \delta b_a^b \\ \delta b_\omega^b \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ C_b^n & 0 & 0 & 0 \\ 0 & C_b^n & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} \nu_a \\ \nu_\omega \\ \nu_{b_a} \\ \nu_{b_\omega} \end{bmatrix} \tag{5.58}$$

the complete output error model is

$$\delta y = \begin{bmatrix} I & 0 & -\left[C_b^n l^b \times\right] & 0 & 0 \\ 0 & 0 & -\left[\left(g^n + \dot{v}_{GPS}^n\right)\times\right] & 0 & 0 \\ 0 & 0 & -\left[h^n \times\right] & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta \rho \\ \delta b_a^b \\ \delta b_\omega^b \end{bmatrix} + \begin{bmatrix} I & 0 & 0 \\ 0 & C_b^n & 0 \\ 0 & 0 & C_b^n \end{bmatrix} \tag{5.59}$$

and the estimated output model is

$$\hat{y} = \begin{pmatrix} \hat{p}^n + \hat{C}_b^n l^b \\ \hat{C}_b^n \tilde{f}^b - \left(\hat{\dot{v}}_{GPS}^n - \hat{C}_b^n \left(\omega_{nb}^b \times \left(\omega_{nb}^b \times l^b\right)\right) - \hat{C}_b^n \left(\dot{\omega}_{nb}^b \times l^b\right)\right) \\ \hat{C}_b^n \tilde{h}^b \end{pmatrix} \tag{5.60}$$

where $\nu_{ta}(t)$ and $\nu_{tm}(t)$ accounts for the total noise that is corrupting the acceleration and magnetic field measurements. Observe that when the UAV is not accelerating or rotating the accelerometers will measure the gravity vector g^n only. However, when the vehicle is either rotating or accelerating, this effect should be subtracted in order to compare the well known gravity vector g^n with its estimate $\hat{g}^n$. The LTI observability matrix is:

$$O = \begin{bmatrix} I & 0 & -\left[C_b^n l^b \times\right] & 0 & 0 \\ 0 & 0 & -\left[\left(g^n + \dot{v}_{GPS}^n\right)\times\right] & 0 & 0 \\ 0 & 0 & -\left[h^n \times\right] & 0 & 0 \\ 0 & I & 0 & 0 & -\left[C_b^n l^b \times\right] C_b^n \\ 0 & 0 & 0 & 0 & -\left[\left(g^n + \dot{v}_{GPS}^n\right)\times\right] C_b^n \\ 0 & 0 & 0 & 0 & -\left[h^n \times\right] C_b^n \\ 0 & 0 & -\left[C_b^n f^b \times\right] & C_b^n & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\left[C_b^n f^b \times\right] C_b^n \\ \hline \multicolumn{5}{c}{0_{101\times 15}} \end{bmatrix} \tag{5.61}$$

Again, regardless of the presence of lever arm, the matrix is still full column rank. To prove O is full column rank, we perform a few row manipulations to obtain its REF:

$$O_{RREF} = \left[\begin{array}{ccccc} I & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & -\left[(g^n + \dot{v}^n_{GPS})\times\right] & 0 & 0 \\ 0 & 0 & -\left[h^n\times\right] & 0 & 0 \\ 0 & 0 & 0 & C^n_b & 0 \\ 0 & 0 & 0 & 0 & -\left[(g^n + \dot{v}^n_{GPS})\times\right] C^n_b \\ 0 & 0 & 0 & 0 & -\left[h^n\times\right] C^n_b \\ \hline \multicolumn{5}{c}{0_{104\times 15}} \end{array}\right] \quad (5.62)$$

Input Output and State Covariance Matrices

The initial state covariance matrix is set to zero (i.e. $P = 0_{15\times 15}$). Allan variance was used in the determination of the input and output covariance matrices. The IMU sensor noise parameters that were found experimentally (Ref. [3]) are presented in Tables 5.1 to 5.9.

Parameter	$Q_{au}\left[\frac{m}{s}\right]$	$N_{au}\left[\frac{\frac{m}{s}}{\sqrt{s}}\right]$	$B_{au}\left[\frac{m}{s^2}\right]$	$K_{au}\left[\frac{\frac{m}{s^2}}{\sqrt{s}}\right]$	$R_{au}\left[\frac{m}{s^3}\right]$
Value	1.94e-4	0.0034	0.0016	7.98e-5	2.9e-4

Table 5.1: Forward Accelerometer Noise Coefficients

Parameter	$Q_{av}\left[\frac{m}{s}\right]$	$N_{av}\left[\frac{\frac{m}{s}}{\sqrt{s}}\right]$	$B_{av}\left[\frac{m}{s^2}\right]$	$K_{av}\left[\frac{\frac{m}{s^2}}{\sqrt{s}}\right]$	$R_{av}\left[\frac{m}{s^3}\right]$
Value	1.76e-4	0.0032	0.0014	1.02e-4	2.62e-6

Table 5.2: Rightward Accelerometer Noise Coefficients

Parameter	$Q_{aw}\left[\frac{m}{s}\right]$	$N_{aw}\left[\frac{\frac{m}{s}}{\sqrt{s}}\right]$	$B_{aw}\left[\frac{m}{s^2}\right]$	$K_{aw}\left[\frac{\frac{m}{s^2}}{\sqrt{s}}\right]$	$R_{aw}\left[\frac{m}{s^3}\right]$
Value	2.9e-4	0.0034	0.0018	1.35e-4	3.51e-6

Table 5.3: Downward Accelerometer Noise Coefficients

Parameter	$Q_{\omega u}\,[rad]$	$N_{\omega u}\left[\frac{rad}{\sqrt{s}}\right]$	$B_{\omega u}\left[\frac{rad}{s}\right]$	$K_{\omega u}\left[\frac{\frac{rad}{s}}{\sqrt{s}}\right]$	$R_{\omega u}\left[\frac{rad}{s^2}\right]$
Value	3.47e-5	7.34e-4	2.07e-4	1.06e-5	2.75e-7

Table 5.4: Forward Gyroscope Noise Coefficients

The input covariance matrix is therefore:

$$Q = \mathrm{diag}\left([\sigma^2_{\nu_{au}}, \sigma^2_{\nu_{av}}, \sigma^2_{\nu_{aw}}, \sigma^2_{\nu_{\omega u}}, \sigma^2_{\nu_{\omega v}}, \sigma^2_{\nu_{\omega w}}, \sigma^2_{\nu^b_{b_{au}}}, \sigma^2_{\nu^b_{b_{av}}}, \sigma^2_{\nu^b_{b_{aw}}}, \sigma^2_{\nu^b_{b_{\omega u}}}, \sigma^2_{\nu^b_{b_{\omega v}}}, \sigma^2_{\nu^b_{b_{\omega w}}}]\right)$$

$Parameter$	$Q_{\omega v}\,[rad]$	$N_{\omega v}\,\left[\frac{rad}{\sqrt{s}}\right]$	$B_{\omega v}\,\left[\frac{rad}{s}\right]$	$K_{\omega v}\,\left[\frac{\frac{rad}{s}}{\sqrt{s}}\right]$	$R_{\omega v}\,\left[\frac{rad}{s^2}\right]$
$Value$	4.33e-5	8.22e-4	2.27e-4	1.19e-5	2.34e-7

Table 5.5: Rightward Gyroscope Noise Coefficients

$Parameter$	$Q_{\omega w}\,[rad]$	$N_{\omega w}\,\left[\frac{rad}{\sqrt{s}}\right]$	$B_{\omega w}\,\left[\frac{rad}{s}\right]$	$K_{\omega w}\,\left[\frac{\frac{rad}{s}}{\sqrt{s}}\right]$	$R_{\omega w}\,\left[\frac{rad}{s^2}\right]$
$Value$	3.48e-5	7.31e-4	1.91e-4	1.47e-5	4.91e-7

Table 5.6: Downward Gyroscope Noise Coefficients

$Parameter$	$Q_{hu}\,[Gauss-s]$	$N_{hu}\,[Gauss-\sqrt{s}]$	$B_{hu}\,[Gauss]$	$K_{hu}\,\left[\frac{Gauss}{\sqrt{s}}\right]$	$R_{hu}\,\left[\frac{Gauss}{s}\right]$
$Value$	3.94e-6	7.39e-5	5.04e-5	2.5e-6	8.34e-8

Table 5.7: Forward Magnetometer Noise Coefficients

$Parameter$	$Q_{hv}\,[Gauss-s]$	$N_{hv}\,[Gauss-\sqrt{s}]$	$B_{hv}\,[Gauss]$	$K_{hv}\,\left[\frac{Gauss}{\sqrt{s}}\right]$	$R_{hv}\,\left[\frac{Gauss}{s}\right]$
$Value$	1.54e-6	6.8e-5	2.83e-5	3.5e-6	1.04e-7

Table 5.8: Rightward Magnetometer Noise Coefficients

$Parameter$	$Q_{hw}\,[Gauss-s]$	$N_{hw}\,[Gauss-\sqrt{s}]$	$B_{hw}\,[Gauss]$	$K_{hw}\,\left[\frac{Gauss}{\sqrt{s}}\right]$	$R_{hw}\,\left[\frac{Gauss}{s}\right]$
$Value$	2.43e-5	1.54e-4	1.36e-4	1.15e-5	3.06e-7

Table 5.9: Downward Magnetometer Noise Coefficients

$$= \text{diag}\left([0.034^2, 0.032^2, 0.034^2, 0.00733^2, 0.0082^2, 0.0073^2, 0.0014^2, 0.0018^2, 0.0023^2, 3.39e^{-8}, 4.28e^{-8}, 6.5e^{-8}]\right) \tag{5.63}$$

The output covariance matrix is composed of GPS, accelerometer and magnetometer sensor noise disturbances. The GPS sensor is corrupted by white noise only. Accelerometer and magnetometer measurements are corrupted by a complex source of noise. Their output noise models are assumed to have a white and Gauss-Markov process noise components. Assuming both type of noises are uncorrelated, the output covariance matrix will be

$$R = \text{diag}\left([\sigma^2_{GPS_N}, \sigma^2_{GPS_E}, \sigma^2_{GPS_D}, \sigma^2_{\nu_{au}}, \sigma^2_{\nu_{av}}, \sigma^2_{\nu_{aw}} + \sigma^2_{\nu_{a_{GM}}}, \sigma^2_{\nu_{hu}}, \sigma^2_{\nu_{hv}}, \sigma^2_{\nu_{hw}} + \sigma^2_{\nu_{h_{GM}}}]\right)$$

$$= \text{diag}\left([0.03^2, 0.03^2, 0.03^2, 0.034^2, 0.032^2 + 0.0024^2, 0.034^2, (7.39e^{-4})^2, (6.8e^{-4})^2, (15e^{-4})^2 + (2.02e^{-4})^2]\right) \tag{5.64}$$

Notice that accelerometer sensor noise is modeled as random walk for state update but is modeled as a Gauss-Markov process noise at the output equation. The reason for this is to avoid an unbounded output covariance matrix for large time ($\sigma^2_{\omega_a} = \left(K_a\sqrt{\frac{3}{T}}\right)^2 t + \sigma^2_{\nu_a}$). The magnetometer sensor noise is modeled as a Gauss - Markov process for the same reason explained above.

EKF Algorithm

The EKF Algorithm proposed is the following:

1. Align and level the UAV.

2. Determine the DC offset bias of magnetometers, gyroscopes and accelerometers.

3. Determine the input noise covariance matrix $Q_{12\times 12}$.

4. Determine the output noise covariance matrix $R_{9\times 9}$.

5. Define the state $\hat{x}$ as:

$$\hat{x} = \begin{bmatrix} \hat{p}^n \\ \hat{v}^n \\ \hat{C}_b^n \\ \hat{b}_a^b \\ \hat{b}_\omega^b \end{bmatrix} \tag{5.65}$$

6. Determine the initial conditions of the state $\hat{x}(0)$. They could assume any value. One alternative is $\hat{p}^n(0) = 0$, $\hat{v}^n(0) = 0$, $\hat{C}_b^n(0) = I_{3\times 3}$, $\hat{b}_a^b(0) = 0$, $\hat{b}_\omega^b(0) = 0$.

7. Determine the initial condition of the predicted covariance matrix $P^-(0)$. One alternative is $P^-(0) = 0_{15\times 15}$.

8. Every time there is a GPS measurement, define the measurement equation as

$$\tilde{y} = \begin{pmatrix} \tilde{p}_{GPS}^n \\ g^n \\ h^n \end{pmatrix} \tag{5.66}$$

9. Every time there is a GPS measurement, subtract the DC offset bias of the gyroscope found in Step 2 from the attitude rate measurement $\tilde{\omega}_{ib}^b$:

$$\tilde{\omega}_{ib}^b = \tilde{\omega}_{ib}^b - \hat{b}_{\omega_{DC}} \tag{5.67}$$

10. Every time there is a GPS measurement, subtract the DC offset bias of accelerometer found in Step 2 from the specific force measurement $\tilde{f}^b$:

$$\bar{f}^b = \tilde{f}^b - \hat{b}_{a_{DC}} \tag{5.68}$$

11. Every time there is a GPS measurement, subtract the DC offset bias of magnetometer found in Step 2 from the magnetic field measurement $\tilde{h}^b$:

$$\bar{h}^b = \tilde{h}^b - \hat{b}_{h_{DC}} \tag{5.69}$$

12. Every time there is a GPS measurement, define the estimated measurement equation as

$$\hat{y} = \begin{pmatrix} \hat{p}^n + \hat{C}_b^n l^b \\ \hat{C}_b^n \bar{f}^b - \left(\dot{\tilde{v}}_{GPS}^n - \hat{C}_b^n \left(\hat{\omega}_{nb}^b \times \left(\hat{\omega}_{nb}^b \times l^b\right)\right) - \hat{C}_b^n \left(\dot{\hat{\omega}}_{nb}^b \times l^b\right)\right) \\ \hat{C}_b^n \bar{h}^b \end{pmatrix} \tag{5.70}$$

13. Every time there is a GPS measurement, define the error as $\delta y = \tilde{y} - \hat{y}$.

14. Every time there is a GPS measurement, define the Output matrix as

$$H = \begin{bmatrix} I & 0 & -\left[\hat{C}_b^n l^b \times\right] & 0 & 0 \\ 0 & 0 & -\left[\left(g^n + \dot{\tilde{v}}_{GPS}^n\right) \times\right] & 0 & 0 \\ 0 & 0 & -\left[h^n \times\right] & 0 & 0 \end{bmatrix} \tag{5.71}$$

15. Every time there is a GPS measurement, update the Kalman gain with

$$K = P^- H^T \left[R + H P^- H^T\right]^{-1} \tag{5.72}$$

16. Every time there is a GPS measurement, correct the predicted state using $\hat{x} = \hat{x}^- + K\delta y$

17. Every time there is a GPS measurement, correct the predicted state covariance matrix using $P = (I - KH) P^-$.

18. Every time there is a GPS measurement, define the state matrix F as

$$F = \begin{bmatrix} 0 & I & 0 & 0 & 0 \\ 0 & -2\Omega_{ie}^n & -\left[\hat{C}_b^n \left(\tilde{f}^b - \hat{b}_a^b\right) \times\right] & \hat{C}_b^n & 0 \\ 0 & 0 & -\Omega_{ie}^n & 0 & \hat{C}_b^n \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.73}$$

19. Every time there is a GPS measurement, define the input noise matrix B as

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \hat{C}_b^n & 0 & 0 & 0 \\ 0 & \hat{C}_b^n & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \tag{5.74}$$

20. Every time there is a GPS measurement, define matrix A as

$$A = \begin{bmatrix} -F & BQB^T \\ 0 & F^T \end{bmatrix} \tag{5.75}$$

21. Every time there is a GPS measurement, compute $\Upsilon = e^{(A \Delta T_{GPS})}$. Then find Φ and Q_d

22. Every time there is a GPS measurement, propagate the predicted covariance matrix as: $P^- = \Phi P \Phi^T + Q_d$

23. Every time there is an IMU measurement, propagate the state using any numerical integration method of the following first order differential equation:

$$\begin{pmatrix} \dot{\hat{p}}^n \\ \dot{\hat{v}}^n \\ \dot{\hat{C}}_b^n \\ \dot{\hat{b}}_a^b \\ \dot{\hat{b}}_\omega^b \end{pmatrix} = \begin{pmatrix} \hat{v}^n \\ \hat{C}_b^n \left(\tilde{f}^b - \hat{b}_a^b\right) + g^n - 2\Omega_{ie}^n \hat{v}^n \\ \hat{C}_b^n \hat{\Omega}_{nb}^b \\ 0 \\ 0 \end{pmatrix} \tag{5.76}$$

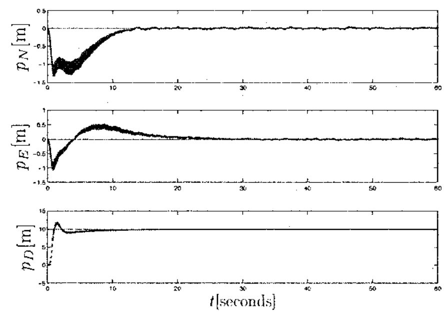
$$\hat{\omega}_{nb}^b = \left(\tilde{\omega}_{ib}^b - \hat{b}_\omega^b\right) - \hat{C}_n^b \omega_{ie}^n \tag{5.77}$$

$$\hat{\Omega}_{nb}^b = \left[\hat{\omega}_{nb}^b \times\right] \tag{5.78}$$

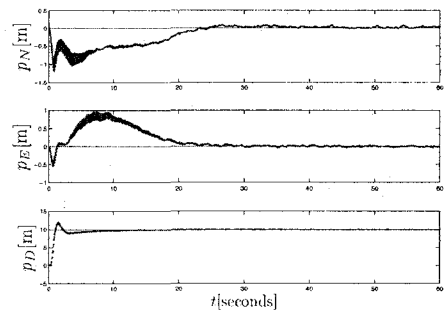
24. Go back to step 8

5.2.3 Simulation Results with and without Lever Arm

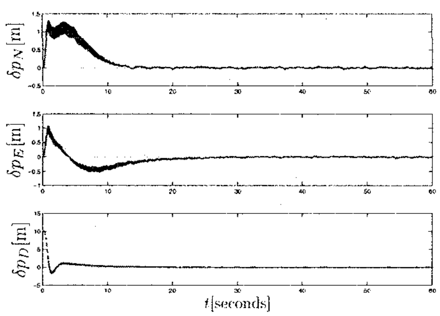
The fifteen states $(\hat{p}^n, \hat{v}^n, \hat{\phi}, \hat{\theta}, \hat{\psi}, \hat{b}_a^b, \hat{b}_\omega^b)$ should converge to their actual values in both cases (with/without lever arm) independent of the UAV trajectory. The system is completely observable if position, magnetic field and acceleration measurements are taken.



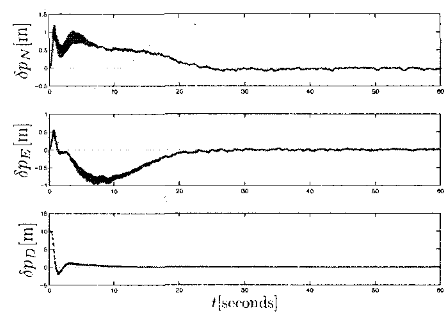
(a) Position states p^n



(b) Position states p^n: lever arm

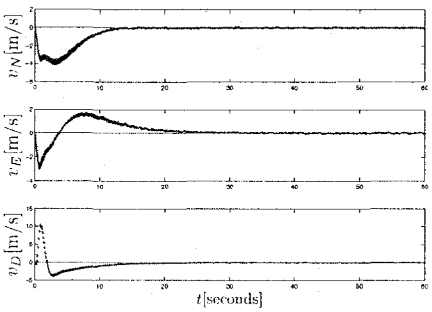


(c) Position error states δp^n

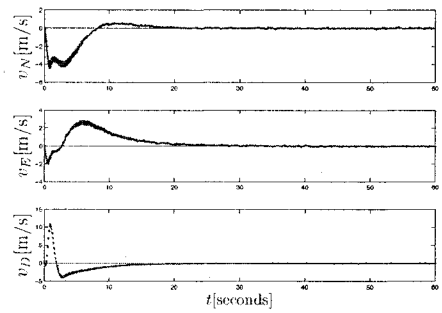


(d) Position error states δp^n: lever arm

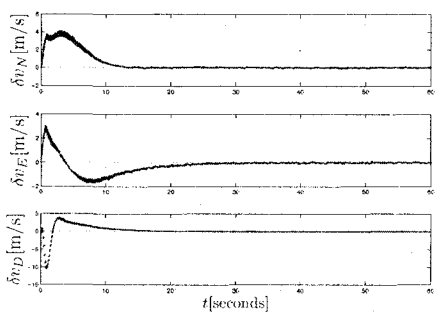
Figure 5.2: Position Estimation



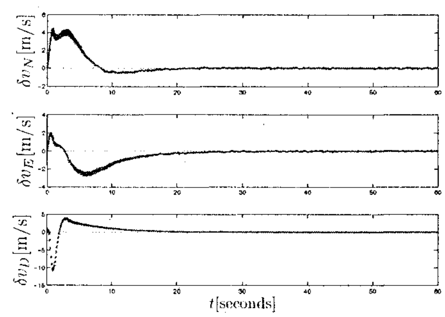
(a) Velocity states v^n



(b) Velocity states v^n: lever arm

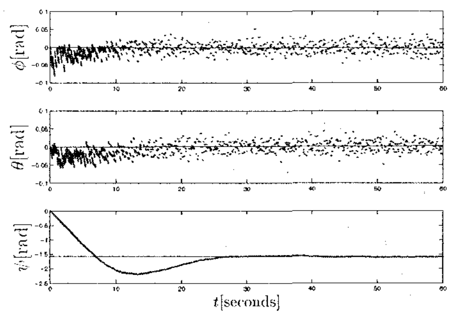


(c) Velocity error states δv^n

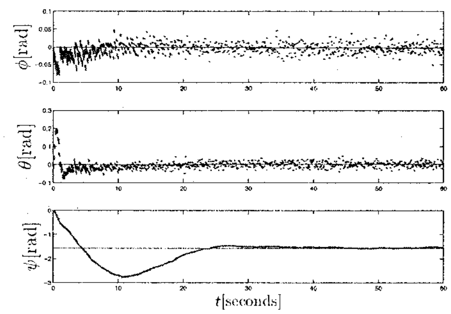


(d) Velocity error states δv^n: lever arm

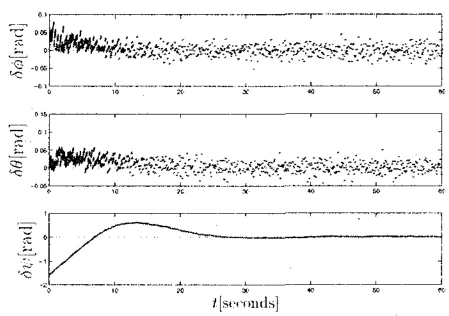
Figure 5.3: Velocity Estimation



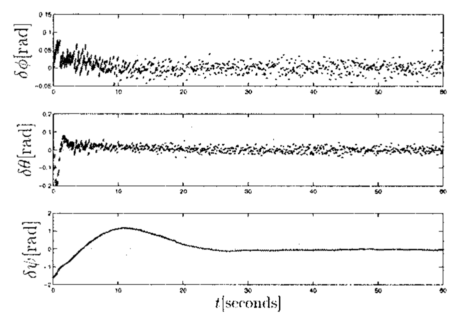
(a) Attitude states ϕ, θ, ψ



(b) Attitude states ϕ, θ, ψ: lever arm

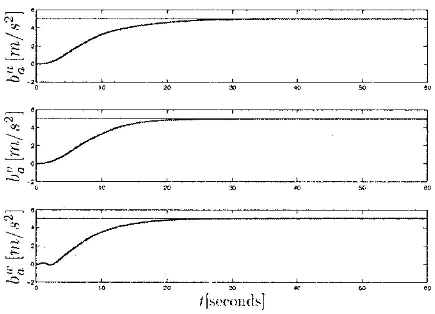


(c) Attitude error states $\delta\rho$

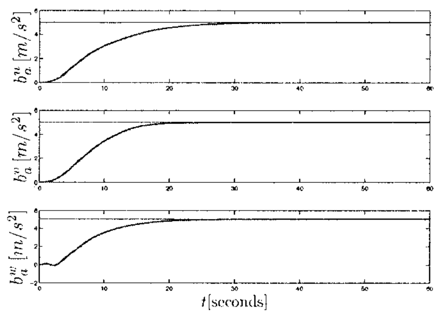


(d) Attitude error states $\delta\rho$: lever arm

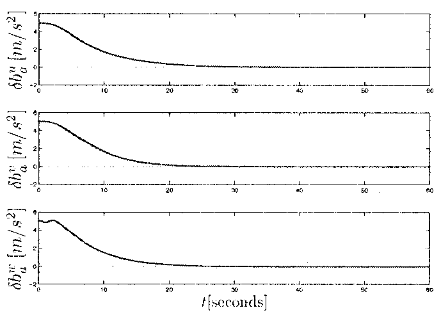
Figure 5.4: Attitude Estimation



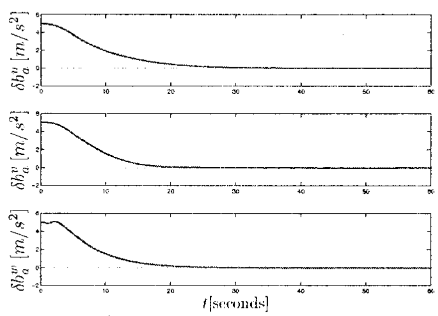
(a) Biases states b_a^b



(b) Biases states b_a^b: lever arm

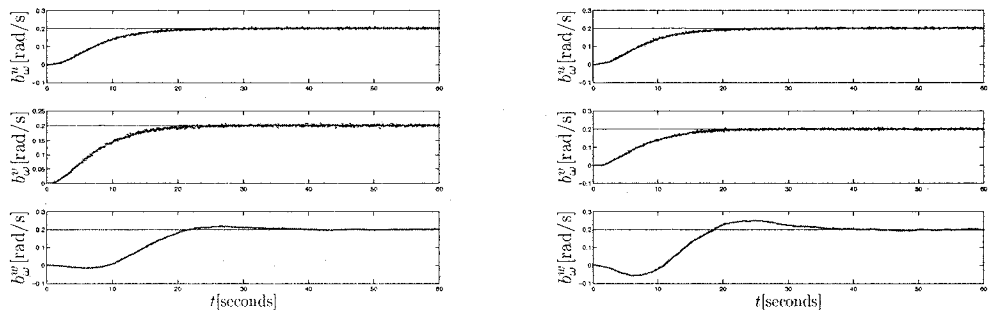


(c) Biases error states δb_a^b



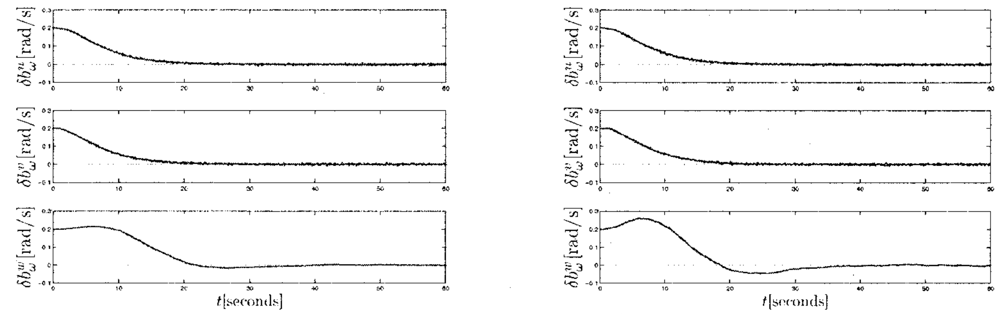
(d) Biases error states δb_a^b: lever arm

Figure 5.5: Accelerometer Bias Estimation



(a) Biases states b^b_ω

(b) Biases states b^b_ω: lever arm



(c) Biases error states δb^b_ω

(d) Biases error states δb^b_ω: lever arm

Figure 5.6: Gyroscope Bias Estimation

Figures 5.2 to 5.6 depict the simulation of the EKF with (right plots) and without (left plots) lever arm.

As can be seen the vehicle was hovering ten meters above the origin. Thus velocity vectors are zero. The roll ϕ and pitch θ angles were zero. The vehicle was heading west $\psi = -\frac{\pi}{2}$. The trajectory parameters are

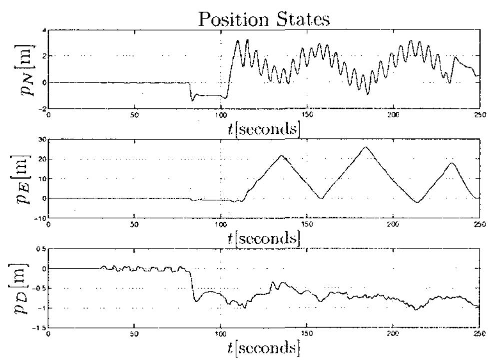
$$l^b = \begin{bmatrix} -0.7 \\ 0 \\ 0 \end{bmatrix} [m],\ p^n = \begin{bmatrix} 0 \\ 0 \\ 10 \end{bmatrix} [m],\ v^n = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \left[\frac{m}{s}\right],\ f^n = \begin{bmatrix} 0 \\ 0 \\ -9.8 \end{bmatrix} \left[\frac{m}{s^2}\right] \quad (5.79)$$

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\frac{\pi}{2} \end{bmatrix} [rad],\ \omega^b_{nb} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \left[\frac{rad}{s}\right] \quad (5.80)$$

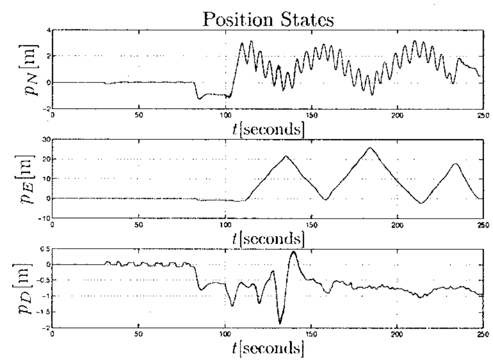
Because the observability matrix in (5.61) is full column rank, it is expected that all states converge to their actual values. There is no difference other than the convergence time of the filter and more pronounced overshoot in the response when lever arm l^b is considered.

Chapter 6

Kalman Filter Results



(a) GPS position states p^n



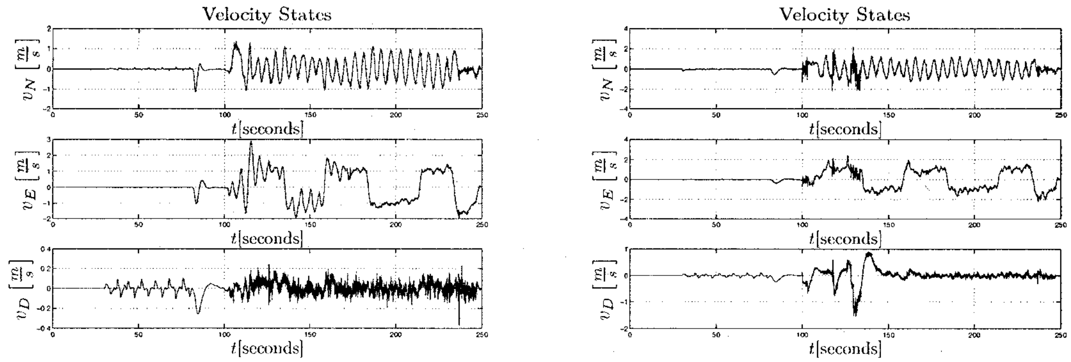
(b) Mag/GPS position states p^n

Figure 6.1: Position Estimation

The Magnetometer/GPS-aided-INS algorithm described in the last section of chapter 5 was tested with real data. These results are going to be compared with the previous GPS-aided-INS EKF design [1]. This will show that when magnetometer and accelerometer samples are included in the measurement equation, the kinematics will always be observable regardless the trajectory the UAV is going through.

Observing Figure 6.1, GPS-aided-INS EKF (left) and Magnetometer/GPS-aided-INS EKF (right) display similar position estimation results. This is expected since the main measurement for position estimation is the GPS report.

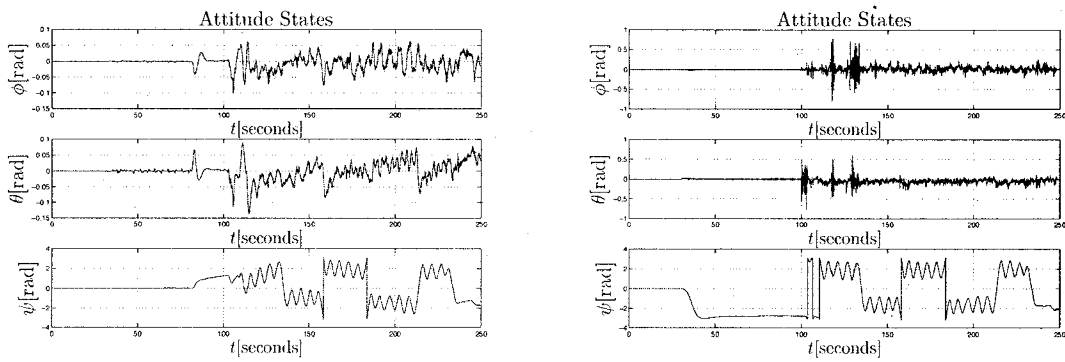
Figure 6.2 shows that the estimation of the velocity states is done better with the GPS-aided-INS design. The particular reason for this is that the Magnetometer/GPS-aided-INS system, is using CDGPS velocity and gy-



(a) GPS velocity states v^n (b) Mag/GPS velocity states v^n

Figure 6.2: Velocity Estimation

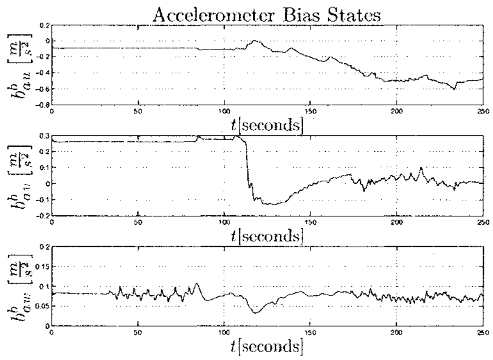
roscope time derivatives in order to estimate the earth's gravity field vector. As it is well known a time derivative will amplify high frequency noise. Nevertheless these time derivatives are needed to guarantee a fully observable estimator.



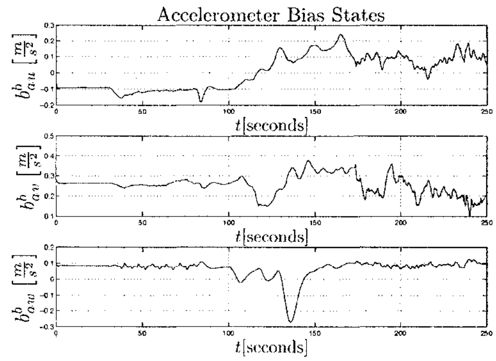
(a) GPS attitude states ϕ, θ, ψ (b) Mag/GPS attitude states ϕ, θ, ψ

Figure 6.3: Attitude Estimation

Similar reasoning applies to the attitude estimation when the vehicle is moving as seen in Figure 6.3. When the vehicle is stationary however, it is interesting to notice that the GPS-aided-INS Kalman estimator is incapable of observing the heading or yaw ψ of the vehicle. Looking at the first hundred seconds of Figure 6.3, the GPS-aided-INS EKF assumes the heading angle is correct as it is initialized. On the other hand with the Magnetometer-GPS-aided-INS, the heading is correctly estimated and equal to -2.88rad. When magnetometer and accelerometer measurements are included in the Kalman Filter, it is expected that all states are estimated correctly independent of the trajectory taken by the vehicle.

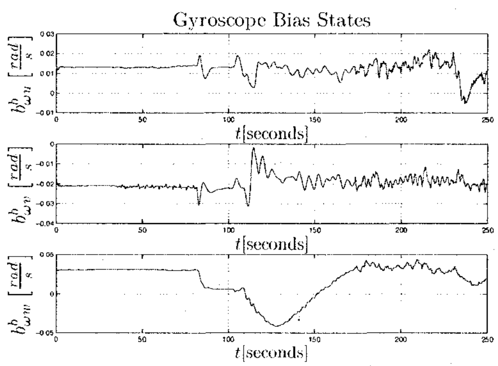


(a) GPS bias states δb_a^b

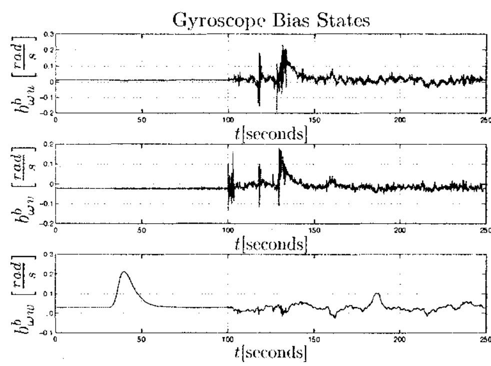


(b) Mag/GPS bias states δb_a^b

Figure 6.4: Attitude Estimation



(a) GPS bias states δb_ω^b



(b) Mag/GPS bias states δb_ω^b

Figure 6.5: Attitude Estimation

The biases are important for the Kalman Filter estimator but have no significance for navigation. The bias estimation of accelerometers and gyroscopes are shown in Figures 6.4 and 6.5 respectively.

The right plots of Figures 6.4 to 6.5 correspond to Magnetometer/GPS-aided-INS testing. The lever arm between the GPS antenna and IMU sensor was assumed equal to $l^b = [0.05, 0, -0.02]$m. Left plots correspond to GPS-aided-INS testing. The lever arm between GPS antenna and IMU sensor was assumed equal to $l^b = [0, 0, 0]$m.

The lever arm distance in this particular cart test is negligible. However, in the helicopter the IMU sensor and GPS antenna will have considerable distance and the lever arm vector in the helicopter's body frame coordinates should be accurately determined.

Chapter 7

Conclusion and Future Work

7.1 Summary of Research and Conclusion

Observability problems of a loosely coupled single antenna GPS-aided-INS navigation were identified. An EKF was used in the estimation of UAV navigation states. It happened that whenever the linearized kinematic model was unobservable the EKF could not estimate correctly some or in the worst case all of the navigation states. The observability analysis presented in Chapter 4 of this thesis stated that with a single GPS antenna: "The UAV's specific force and its first and second order time derivatives (*i.e.* f^n, $\dot{f}^n$, $\ddot{f}^n$) should be linearly independent" as a sufficient condition for the UAV's trajectories to completely estimate the fifteen navigation states with an EKF. If the mentioned condition was not satisfied, the null space basis of the observability rank test matrix determined in which direction the linearized kinematic model was unobservable and which states will be affected the most.

To overcome the problem of unobservable trajectories, the measurement equation was augmented from a three to a nine dimensional vector. The measurement vector included the Earth's magnetic field and the Earth's gravitational field. Unlike the position vector, which was measured by the GPS system, the Earth's magnetic and gravitational fields were known but unmeasured. This augmentation rendered the linearized fifteen state navigation problem completely observable.

To estimate the magnetic and gravitational fields of earth, the IMU's magnetometers and accelerometers were used in conjunction with the GPS velocity measurement and its time derivative. The necessary condition for complete observability of the Magnetometer-GPS-aided-INS setup is that the magnetometer and accelerometer measurements must not have DC offset bias. Off-line determination of these DC offset bias remedies the observability issue.

To prove the augmented output filter works, an EKF was simulated, implemented and tested with real data. A comparison between the loosely coupled GPS-aided-INS and Magnetometer-GPS-aided-INS Kalman was presented and discussed in Chapter 7.

7.2 Future Work

The EKF used a loosely coupled GPS system. The main reason for this is that the provider NovAtel has invested time in developing reliable GPS systems and its sensor must provide reliable position and velocity data. It will be time consuming though interesting to develop a tightly coupled Magnetometer-GPS-aided-INS.

It is assumed that the Earth's magnetic field model is constant in the region where the UAV is flying. If the UAV is going to fly large distances, a new magnetic field vector must be found. The Earth's magnetic model presented in Appendix B can be used to determine the magnetic field at any location, however it is computationally expensive and probably not practical to implement. For real time applications it would be more practical to use a look up table.

Bibliography

[1] David Kastelan. Loosely coupled 6-DOF GPS-aided INS by Kalman filtering. Technical report, ANCL, University of Alberta, 2007.

[2] Sinpyo Hong, Man Hyung Lee, Ho-Hwan Chun, Sun-Hong Kwon, and Jason L. Speyer. Observability of error states in GPS/INS integration. *IEEE Transactions on Vehicular Technology*, 54:731–743, Mar. 2005.

[3] David Kastelan. Loosely coupled 3-DOF GPS-aided INS by Kalman filtering. Technical report, ANCL, University of Alberta, 2007.

[4] Jay A. Farrell and Matthew Barth. *The Global Positioning System & Inertial Navigation*. McGraw-Hill, New York, NY, 1999.

[5] Chi Tsong Chen. *Linear System Theory and Design*. Oxford, 1999.

[6] Thomas Stork. Electronic Compass Design using KMZ51 and KMZ52. Technical report, Phillips Semiconductors, Systems Laboratory Hamburg, Germany, March 2000.

[7] Microstrain. 3DM-GX1 HardIronCalibration. Technical report, Microstrain, Inc, 2005.

[8] Giri Baleri. AHRS 400 Series Installation and Troubleshooting Guide. Technical report, Crossbow Technology, Inc, 2005.

[9] Demoz Gebre Egziabher, Roger C. Hayward, and J. David Powell. Design of Multisensor Attitude Determination Systems. *IEEE Transactions On Aerospace and Electronic Systems*, 40:627 – 649, April 2004.

[10] Demoz Gebre-Egziabher. *Design and Performance Analysis of a Low Cost Aided Dead Reckoning Navigator.* PhD thesis, Stanford University, 2001.

[11] Yunchun Yang and Jay A. Farrell. Two Antennas GPS-Aided INS for Attitude Determination. *IEEE Transactions on Control Systems Technology*, 11:905–918, 2003.

[12] Yunchun Yang. *Tightly Integrated Attitude Determination Methods for Low-Cost Inertial Navigation: Two-Antenna GPS and GPS/Magnetometer.* PhD thesis, University of California Riverside, 2005.

[13] D. Goshen-Meskin and I.Y. Bar-Itzhack. Observability Analysis of Piece Wise Constant Systems Part II: Application to Inertial Navigation In-Flight Alignment. *IEEE Transactions on Aerospace and Electronic Systems*, 28:1068–1075, 1992.

[14] Robert G. Brown and Patric Y.C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering.* Wiley, 1997.

[15] Wilson J. Rugh. *Linear System Theory.* Prentice Hall, 1996.

[16] Susan Macmillan. *International Geomagnetic Reference Field.* IAGA, [Online] Available http://www.ngdc.noaa.gov/IAGA/vmod/igrf.html, March 2005.

[17] TheMathWorks. *Associated Legendre Functions.* [Online] Available http://www.mathworks.com/access/helpdesk/help/techdoc/index.html?/. access/helpdesk/help/techdoc/ref/legendre, January 2008.

[18] Novatel. *OEM4 Family of Receivers - Command and Log Reference Manual.* Novatel, 2005.

[19] Wikipedia. *IEEE 754 - 1985.* Wikipedia, [Online] Available http://en.wikipedia.org/wiki/IEEE_754, January 2008.

[20] Wikipedia. *Binary Numeral System.* Wikipedia, [Online] Available http://en.wikipedia.org/wiki/Binary_numeral_system, February 2008.

[21] Novatel. *OEM4 Family of Receivers Installation & Operation Manual.* Novatel, 2005.

[22] Susan Mclean. *Compute Earth's Magnetic Field Values.* National Geophysical Data Center (NGDC), [Online] Available http://www.ngdc.noaa.gov/seg/geomag/magfield.shtml, February 2008.

[23] F. Landis Markley and Itzhack-Y. Bar-Itzhack. Unconstrained optimal transformation matrix. *IEEE Transactions on Aerospace and Electronic Systems*, 34:338–340, January 1998.

[24] D. Li, J. Wang, and S. Babu. Nonlinear Stochastic Modeling for INS Derived Doppler Estimates in Ultra-Tight GPS/PL/INS Integration.

[25] Niklas Hjortsmarker. Experimental System for Validating GPS/INS Integration Algorithms. Master's thesis, Lulea University of Technology, 2005.

Appendix A

Notation, Symbols, Acronyms and Useful Formulas

A.1 Notation

1. Any translational vector is expressed as: f^b. The superscript "b" is the coordinate frame where the vector is decomposed, in this case the body frame.

2. Any rotational vector is expressed as: ω_{ib}^b. The subscript "ib" means the rotation of frame "b" with respect to frame "i", the superscript "b" again is the coordinate frame where the vector is decomposed.

3. Any vector measurement has a "tilde" sign. E.g. $\tilde{f}^b$ means specific force vector measured in coordinate frame "b". E.g. $\tilde{\omega}_{ib}^b$ means the relative rotation of frame "b" with respect to frame "i" measured in the coordinate frame "b".

4. Any vector estimation has a "hat" sign. E.g. $\hat{p}^n$ means the position vector estimation in coordinate frame "n".

5. The sensor that solves a physical measurement is referenced with a subscript. E.g. The GPS position measurement is $\tilde{p}_{GPS}^n$. E.g. The accelerometer sensor bias is b_a^b where "a" stands for "accelerometer". E.g. The gyroscope sensor bias is b_ω^b and "ω" stands for "gyroscope".

6. Any vector component is referenced with a subscript. i.e. b^b_{au} means the "u^{th}" component of vector b^b_a.

7. The notation Ω^b_{nb} is the skew symmetric matrix form of ω^b_{nb} and is used to express vector cross-products. For e.g.:

$$\omega^b_{nb} \times l^b = \Omega^b_{nb} l^b = \begin{bmatrix} 0 & -\omega^b_{nbw} & \omega^b_{nbv} \\ \omega^b_{nbw} & 0 & -\omega^b_{nbu} \\ -\omega^b_{nbv} & \omega^b_{nbu} & 0 \end{bmatrix} \begin{bmatrix} l^b_u \\ l^b_v \\ l^b_w \end{bmatrix} \tag{A.1}$$

8. The notation $\lfloor \omega^b_{nb} \times \rfloor$ is equivalent to Ω^b_{nb}. i.e. is another way to represent the skew symmetric form of a vector.

A.2 Symbols

The coordinate frame symbols used in this thesis were:

Symbol	*Frame*
b	Body frame coordinates
n	Navigation frame coordinates
t	Fixed tangent plane coordinates
e	ECEF coordinates

Table A.1: Coordinate Frames

The main variable symbols used in this thesis were:

Symbol	*Description*
p^n	Position in navigation frame
v^n	Velocity in navigation frame
C_b^n	Direction Cosine Matrix. Transforms vectors from body frame to navigation frame
C_n^b	Direction Cosine Matrix. Transforms vectors from navigation frame to body frame
b_a^b	Accelerometer bias
b_ω^b	Gyroscope bias
P	Actual State covariance matrix
Q	Input covariance matrix
R	Output covariance matrix
P^-	Predicted state covariance matrix
Ω_{nb}^b	Relative rotation of body frame with respect to navigation frame decomposed in body frame coordinates
Ω_{ib}^b	Relative rotation of body frame with respect to the inertial frame decomposed in body frame coordinates
Ω_{ie}^n	Relative rotation of the earth frame with respect to the intertial frame decomposed in navigation frame coordinates
$[h^n\times]$	Magnetic field vector decomposed in navigation frame coordinates
F	State transition matrix
B	Input to state matrix
H	State to output matrix
l^b	Lever arm vector decomposed in body frame coordinates
$[\delta\rho\times]$	Attitude error vector of Direction Cosine Matrix (DCM) C_b^n

Table A.2: Variable Definitions

A.3 Acronyms

The following is a list of acronyms used in this thesis:

Acronym	*Definition*
GPS	Global Positioning System
DGPS	Differential-GPS
CDGPS	Carrier phase Differential - GPS
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
DCM	Direction Cosine Matrix
UAV	Unmanned Aerial Vehicles
REF	Row Echelon Form (Matrices)
RREF	Reduced Row Echelon Form (Matrices)
3DOF	Three Degrees of Freedom
6DOF	Six Degrees of Freedom
EKF	Extended Kalman Filter
LTI	Linear Time Invariant
LTV	Linear Time Varying

Table A.3: Acronyms

A.4 Useful Formulas

1. Gyroscopes are inertial sensors that measure rotation rate of body with respect to inertial frame. The true rotation rate can be expressed as:

$$\omega_{ib}^{b} = \underbrace{\tilde{\omega}_{ib}^{b}}_{\text{measurement}} + b_{\omega}^{b} + \nu_{\omega} \tag{A.2}$$

2. Accelerometers are inertial sensors that measure specific force. The true specific force can be expressed as:

$$f^{b} = \underbrace{\tilde{f}^{b}}_{\text{measurement}} + b_{a}^{b} + \nu_{a} \tag{A.3}$$

3. C_b^n is a DCM that rotates vectors from body frame coordinates to navigation frame coordinates. Roll (ϕ), pitch (θ) and yaw (ψ) are computed from C_b^n using the following formula[4, Ch. 2.5.3.1]:

$$\begin{aligned}
\phi &= \arctan 2(C_b^n[3,2], C_b^n[3,3]) \\
\theta &= \arctan\left(\frac{C_b^n[3,1]}{\sqrt{1-(C_b^n[3,1])^2}}\right) \\
\psi &= \arctan 2(C_b^n[2,1], C_b^n[1,1])
\end{aligned} \tag{A.4}$$

4. The autocorrelation of a stochastic signal is[14, Ch. 2.5]:

$$R_x(t_1, t_2) = E[x(t_1)x(t_2)] \tag{A.5}$$

5. The Power Spectral Density function of a stochastic signal is[14, Ch. 2.7]:

$$S_x(j\omega) = \int_{-\infty}^{+\infty} R_x(\tau)e^{-j\omega\tau}d\tau \tag{A.6}$$

6. The mean square value of a stochastic signal is[14, Ch. 2.7]:

$$E[x^2] = \frac{1}{2\pi}\int_{-\infty}^{+\infty} S_x(j\omega)d\omega = \frac{1}{2\pi j}\int_{-j\infty}^{j\infty} S_x(s)ds \tag{A.7}$$

7. The autocorrelation function of a linear filtered stochastic signal $f(t)$ is[14, Ch. 3.8]:

$$R_x(t_1, t_2) = \int_0^{t_1}\int_0^{t_2} g(u)g(v)R_f(u - v + t_2 - t_1)dudv \tag{A.8}$$

8. The mean square value of a linear filtered stochastic signal $f(t)$ is[14, Ch. 3.8]:

$$E[x^2] = \int_0^{t}\int_0^{t} g(u)g(v)R_f(u - v)dudv \tag{A.9}$$

9. The mean square value of a Gauss-Markov process is:

$$E[x^2] = \frac{\sigma_\omega^2\tau}{2}\left(1 - e^{-\frac{2t}{\tau}}\right) \tag{A.10}$$

10. The mean square value of a Random-Walk process is:

$$E[x^2] = \sigma_\omega^2 t \tag{A.11}$$

Appendix B

Observability Test

Given a linear dynamic system in state space form

$$\dot{x}(t) = F(t)x(t) + G(t)u(t) \tag{B.1}$$

$$y = H(t)x(t) + D(t)u(t) \tag{B.2}$$

We define an observability matrix

$$O = \begin{bmatrix} N_0 \\ N_1 \\ N_2 \\ \vdots \\ N_{n-1} \end{bmatrix} \tag{B.3}$$

where

$$N_0(t) = H(t) \tag{B.4}$$

$$N_{k+1}(t) = N_k(t)F(t) + \frac{dN_k(t)}{dt}, \ 0 \leq k \leq n-1 \tag{B.5}$$

Three types of observability exist when analyzing LTV systems. Complete observability, differential observability and instantaneous observability. Complete observability over an interval $[t_0, t_1]$ requires the rank of the matrix O is n for some $t \in [t_0, t_1]$. Differential observability on $[t_0, t_1]$ requires the rank of O is n for any subinterval in $[t_0, t_1]$. Instantaneous observability states that the rank of matrix O is n for every $t \in [t_0, t_1]$ [15, Ch. 11]. Evidently, instantaneous observability is the strongest observability condition. For LTI case the three criterions are equivalent.

Appendix C

Earth's Magnetic Field

In source free regions, the magnetic field of the earth is modeled as the negative gradient of a scalar potential V which is represented by a truncated series expansion[16]:

$$V(r,\theta,\lambda,t) = \sum_{n=1}^{N_{max}} \left(\frac{R}{r}\right)^{n+1} \sum_{m=0}^{n} \left(g_n^m(t)\cos(m\lambda) + h_n^m(t)\sin(m\lambda)\right) S_m^n(\theta) \tag{C.1}$$

Where "r" is the distance from the center of the earth, "R" is the reference radius (R = 6371.2 km). The term "θ" is the colatitude (i.e. 90^0 minus latitude).

The terms $g_n^m(t)$ and $h_n^m(t)$ are coefficients as a function of time and are tabulated in reference[16]. This model assumes the coefficients variation (*i.e.* $g_n^m(t), h_n^m(t)$) are linear in time within a 5 year interval. The last year these coefficients were determined was 2005.

Generally the magnetic potential model for V is truncated to degree ten (i.e. $N_{max} = 10$). The $P_m^n(\theta)$ terms are the Schmidt semi-normalize associated Legendre functions of degree n and order m.
The Legendre polynomial of degree n is[17]:

$$S_n(\theta) = \frac{1}{2^n n!}\left[\frac{d^n}{d\theta^n}\left(\theta^2 - 1\right)^n\right] \tag{C.2}$$

The Associated Legendre function of (C.2) is:

$$S_n^m(\theta) = (-1)^m \left(1-\theta^2\right)^{\frac{m}{2}} \frac{d^m}{d\theta^m} S_n(\theta) \tag{C.3}$$

It is difficult to calculate the gradient of the scalar function "V" since it involves expressing the cartesian coordinates $[x, y, z]$ in terms of the curvilinear coordinates $[r, \theta, \lambda]$.

The radius (r) is computed with the position in ECEF coordinates of the vehicle as follows:

$$r = \sqrt{x^2 + y^2 + z^2} \tag{C.4}$$

The longitude (λ) is computed with the vehicle's position in ECEF coordinates as follows:

$$\lambda = \arctan 2(y, x) \tag{C.5}$$

It is more involved to compute latitude. The semi-major axis of the WGS84 ellipse is $a = 6378137$m. The eccentricity of the ellipsoid is $e = 8.1819190842622 \times 10^{-2}$. The semi-minor axis of the WGS84 ellipse is thus $b = \sqrt{a^2\left(1-e^2\right)}$.

The following coefficients are used:

$$e' = \sqrt{\left(\frac{a}{b}\right)^2 - 1} \tag{C.6}$$

$$p = \sqrt{x^2 + y^2} \tag{C.7}$$

$$\alpha = \arctan 2(a * z, b * p) \tag{C.8}$$

The co-latitude is computed as follows:

$$\theta = \frac{\pi}{2} - \arctan \frac{z + e'^2 b \sin^3(\alpha)}{p - e^2 a \cos^3(\alpha)} \tag{C.9}$$

Now that the curvilinear coordinates (r, λ, θ) are expressed as a function of the cartesian coordinates (x, y, z), the gradient of the magnetic potential is:

$$h^e = \nabla V = \begin{bmatrix} \frac{\partial}{\partial x} V(x, y, z) \\ \frac{\partial}{\partial y} V(x, y, z) \\ \frac{\partial}{\partial z} V(x, y, z) \end{bmatrix} \tag{C.10}$$

The magnetic field found in equation (C.10) is expressed in ECEF coordinates. To transform it into local tangent plane the following equation is used:

$$h^n = \begin{bmatrix} -\sin\lambda\cos\phi & -\sin\lambda\sin\phi & \cos\lambda \\ -\sin\phi & \cos\phi & 0 \\ -\cos\lambda\cos\phi & -\cos\lambda\sin\phi & -\sin\lambda \end{bmatrix} h^e \tag{C.11}$$

Appendix D

Novatel Commands

There are two stations which are the Base Station and the Rover Station. The purpose is to exploit the precision of the Novatel Devices. Setting up the "Novatel ProPak G2plus" as the base station and the "Novatel FlexPak" as the rover station by configuring these devices properly and the whole Carrier phase Differential GPS (a.k.a. CDGPS) system will report position with accuracies as good as three centimeters.

Binary format is the preferred communication method with both receivers (i.e. Base Station and Rover Station) mainly because the throughput of data is the minimum, allowing more time to process data between consecutive data packets that arrive to the interpreter and finally to the Kalman Filter.

There are two types of commands that can be written to the Base Station's and Rover Station's devices. The first type of commands is for device configuration. The second type of commands is to request information from the device such as position, velocity or any other GPS data.

This chapter is going to be divided into: Header of Messages, Base Station Messages and Rover Station Messages.

D.1 Header of Messages

Refer to the device manual [18] for further information on the matter. All binary messages must have headers for establishing communication with either the Base Station or the Rover Station. All headers have the same length which is 28 bytes. The header itself has 16 fields, and each field has a fixed size in bytes which added together result in 28 bytes. There are four fields that together indicate the beginning of a new message and are at the very beginning of the header.

Table D.1 shows a description of every single field that belongs to the header:

Field Number	*Field Name*	*Field Type*	*Description*	*Binary Bytes*
1	Sync	Char	Hexadecimal 0xAA	1
2	Sync	Char	Hexadecimal 0x44	1
3	Sync	Char	Hexadecimal 0x12	1
4	Header Length	Uchar	Length of header	1
5	Message ID	Ushort	ID number of the command	2
6	Message Type	Char	0x00: Binary Message	1
7	Port Address	Uchar	Which port is the processor connected to	1
8	Message Length	Ushort	Length in bytes of the body of message This does not include the header nor the CRC	2
9	Sequence	Ushort	Multiple related logs. Most logs only come out one at a time in which case this number is 0	2
10	Idle Time	Uchar	Time processor is idle in the last second between successive logs with the same message ID	1
11	Time Status	Enum	Quality of GPS Time	1
12	Week	Ushort	GPS week number	2
13	Milliseconds	GPSec	Milliseconds from beginning of GPS week	4
14	Receiver Status	Ulong	32 bits representing the status of various hardware and software components	4
15	Reserved	Ushort	Reserved for internal use	2
16	Receiver S/W Version	Ushort	software build number	2

Table D.1: Header of Binary Messages

The most important fields to be careful about when writing to the Base or Rover stations are the 5th field (i.e. Message ID), the 6th field (i.e. Message type), the 7th field (i.e. Port Address) and the 8th field (i.e. Message Length). Other fields can be written with 0 as the hexadecimal value. The Message ID is the hexadecimal conversion of the ID number that is given in each of the commands [18, Ch. 2.6] and each of the logs [18, Ch. 3.4]. *Log* itself is a command with message ID equal to one (i.e. 0x01). It logs any response message made by the receiver (Base or Rover) which are sent to the Navigation Estimator.

D.2 Base Station Messages

The following set of messages need to be written in the same order to configure the Base Station:

1. Com com2 19200 n 8 1 n off off.
2. Interfacemode com2 none rtca off.
3. Fix position *latitude longitude height.*
4. Log com2 rtcaobs ontime 1.
5. Log com2 rtcaref ontime 10.
6. Log com2 rtca1 ontime 10 3.
7. Log com2 rtcaephem ontime 10 7.

D.2.1 "Com" Command

The "Com" command header in binary format is presented in Table D.2:

Hexadecimal Command	*Description*
0xAA	Beginning and present in all Headers
0x44	Beginning and present in all Headers
0x12	Beginning and present in all Headers
0x1C	Total length of Header
0x04,0x00	Command ID for *Com*
0x00	Means the message is going to be in binary format
0xC0	Means the processor is connected to "This port" of the receiver
0x20,0x00	Length of the message not taking into account the header length nor the 4 extra bytes of the CRC32 checksum
0x00,0x00	Sequence number for multiple related *Logs* to be filled in with zeros
0x00	Idle time in the last second between successive logs with same message ID, to be filled in with zeros
0x00	Time status of GPS, because is a request from the processor it can be filled in with zeros
0x00,0x00	GPS week number, not to be considered when requesting a log
0x00,0x00,0x00,0x00	Milliseconds from beginning of GPS week
0x00,0x00,0x00,0x00	Receiver Status ignored when writing data
0x00,0x00	For internal use of the device ignored when writing
0x00,0x00	Software version ignored when writing

Table D.2: Header of "Com" command

Just after the header the body of the message must be written, its values are shown in Table D.3:

Hexadecimal Command	*Description*
0x40,0x00,0x00,0x00	Port of the receiver that is going to be configured This case is "com2"
0x00,0x4B,0x00,0x00	Communication baud rate (bps). This case is 19200bps
0x00,0x00,0x00,0x00	Parity of message. This case it has no parity
0x08,0x00,0x00,0x00	Number of databits. This case it has 8 bits
0x01,0x00,0x00,0x00	Number of stop bits. This case is 1 stop bit
0x00,0x00,0x00,0x00	Handshake. This case it has no handshake
0x00,0x00,0x00,0x00	Echo. 0 value means "off"
0x00,0x00,0x00,0x00	Break. 0 value means "off"

Table D.3: Body of "Com" command

All messages should end with a checksum. The protocol of the checksum is the CRC32. When Header and Body are stacked together the CRC32 checksum is: **0x24,0x97,0xF8,0xA0**. The "Com" command message must have in the same order the "Header" then the "Body" and the CRC32 checksum just computed.

D.2.2 "Interfacemode" Command

The "Interfacemode" command header in binary format is presented in Table D.4:

Hexadecimal Command	*Description*
0xAA	Beginning and present in all Headers
0x44	Beginning and present in all Headers
0x12	Beginning and present in all Headers
0x1C	Total length of Header
0x03,0x00	Command ID for *Interfacemode*
0x00	Means the message is going to be in binary format
0xC0	Means the processor is connected to "This port" of the receiver
0x10,0x00	Length of the message not taking into account the header length nor the 4 extra bytes of the CRC32 checksum
0x00,0x00	Sequence number for multiple related *Logs* to be filled in with zeros
0x00	Idle time in the last second between successive logs with same message ID, to be filled in with zeros
0x00	Time status of GPS, because is a request from the processor it can be filled in with zeros
0x00,0x00	GPS week number, not to be considered when requesting a log
0x00,0x00,0x00,0x00	Milliseconds from beginning of GPS week
0x00,0x00,0x00,0x00	Receiver Status ignored when writing data
0x00,0x00	For internal use of the device ignored when writing
0x00,0x00	Software version ignored when writing

Table D.4: Header of "Interfacemode" command

The body of the "Interfacemode" command is the following:

Hexadecimal Command	*Description*
0x40,0x00,0x00,0x00	Port of the receiver that is going to be configured This case is "com2"
0x00,0x00,0x00,0x00	Reception type. This case is None
0x03,0x00,0x00,0x00	Transmission type. This case is RTCA
0x00,0x00,0x00,0x00	Response Generation. This case it is OFF

Table D.5: Body of "Interfacemode" command

The CRC32 checksum of header and body is: **0x0A,0xDF,0x44,0x23**.

D.2.3 "Fix" Command

The position of the Base Station antenna must be fixed in order for the system to operate in CDGPS mode. If the geodetic coordinates of the place of base station operation is known, fix the position to that particular value. Like in most cases this position is unknown and must be requested to the GPS base station. The accuracy of the measurement will be at most 3m because the position computation is single point type[18, Ch. 3.4.3].

The GPS processor's manufacturers strongly recommend that the position entered to the command "Fix position" be good to within few meters. In case the position is unknown, the Base should request the position of the "Base station" antenna. A high accuracy in the position computation is obtained once more than six satellites are being tracked[18, Ch. 2.6.21]. Let's assume for now that the position is known to within the specifications. An algorithm will be given in the section "Fix Position Algorithm" that fixes the "Base station" position by software at any location.

Assuming for now the "Base station" antenna is positioned at $(\lambda, \phi, h) = (53.525950^0, -113.528049^0, 690.434756[\mathrm{m}])$, the header of the "Fix position 53.525950^0 -113.528049^0 690.434756" command is shown in Table D.6 and its body is shown in Table D.7.

For the body of the message, it is necessary to transform the decimal numbers of latitude (λ), longitude (ϕ) and height (h) into the binary numerical standard IEEE-754. References [19] and [20] provide an explanation on how to convert decimal numbers into the IEEE standard and vice-versa.

Hexadecimal Command	*Description*
0xAA	Beginning and present in all Headers
0x44	Beginning and present in all Headers
0x12	Beginning and present in all Headers
0x1C	Total length of Header
0x2C,0x00	Command ID for *Fix*
0x00	Means the message is going to be in binary format
0xC0	Means the processor is connected to "This port" of the receiver
0x1C,0x00	Length of the message not taking into account the header length nor the 4 extra bytes of the CRC32 checksum
0x00,0x00	Sequence number for multiple related *Logs* to be filled in with zeros
0x00	Idle time in the last second between successive logs with same message ID, to be filled in with zeros
0x00	Time status of GPS, because is a request from the processor it can be filled in with zeros
0x00,0x00	GPS week number, not to be considered when requesting a log
0x00,0x00,0x00,0x00	Milliseconds from beginning of GPS week
0x00,0x00,0x00,0x00	Receiver Status ignored when writing data
0x00,0x00	For internal use of the device ignored when writing
0x00,0x00	Software version ignored when writing

Table D.6: Header of "Fix" command

Hexadecimal Command	*Description*
0x03,0x00,0x00,0x00	Means that the Fix type is position i.e. "Fix position"
0x64,0xAA,0x60,0x54 0x52,0xC3,0x4A,0x40	Latitude position data
0xDE,0x6B,0x08,0x8E 0xCB,0x61,0x5C,0xC0	Longitude position data
0xEA,0x8D,0x5A,0x61 0x7A,0x93,0x85,0x40	Altitude position data

Table D.7: Body of "Fix" command

Latitude, longitude and height fields are already given in big endian format. The CRC32 checksum of the latter message is: **0xE9,0xA8,0xB6,0xC7.**

D.2.4 "Log" Command

The most important command to the user point of view. It can request any type of GPS data information of the antenna. In the base station the list of requests are:

1. Log com2 rtcaobs ontime 1

2. Log com2 rtcaref ontime 10

3. Log com2 rtca1 ontime 10 3

4. Log com2 rtcaephem ontime 10 7

The header for all four messages is presented in Table D.8:

Hexadecimal Command	*Description*
0xAA	Beginning and present in all Headers
0x44	Beginning and present in all Headers
0x12	Beginning and present in all Headers
0x1C	Total length of Header
0x01,0x00	Command ID for *Log*
0x00	Means the message is going to be in binary format
0xC0	Means the processor is connected to "This port" of the receiver
0x1C,0x00	Length of the message not taking into account the header length nor the 4 extra bytes of the CRC32 checksum
0x00,0x00	Sequence number for multiple related *Logs* to be filled in with zeros
0x00	Idle time in the last second between successive logs with same message ID, to be filled in with zeros
0x00	Time status of GPS, because is a request from the processor it can be filled in with zeros
0x00,0x00	GPS week number, not to be considered when requesting a log
0x00,0x00,0x00,0x00	Milliseconds from beginning of GPS week
0x00,0x00,0x00,0x00	Receiver Status ignored when writing data
0x00,0x00	For internal use of the device ignored when writing
0x00,0x00	Software version ignored when writing

Table D.8: Header of "Log" command

The body of the first message looks like in Table D.9:

Hexadecimal Command	*Description*
0x40,0x00,0x00,0x00	This log is loaded at Com2 of "Base Station" device
0x06,0x00	Message ID for "rtcaobs" log
0x00	Means that the message type is in binary format
0x00	This field is reserved and can be filled in with zeros
0x02,0x00,0x00,0x00	Means the log is going to be triggered ("ontime")
0x00,0x00,0x00,0x00 0x00,0x00,0xF0,0x3F	The period of time the log is going to be trigged is 1 second
0x00,0x00,0x00,0x00 0x00,0x00,0x00,0x00	Offset time of message when logged
0x00,0x00,0x00,0x00	This log can be removed by "Unlogall" command

Table D.9: Body of "Log rtcaobs" Message

The second message's body will look like in Table D.10. Table D.11 show the body of third message. In Table D.12 it is shown the body of the fourth and last message.

Hexadecimal Command	*Description*
0x40,0x00,0x00,0x00	This log is loaded at Com2 of "Base Station" device
0x0B,0x00	Message ID for "rtcaref" log
0x00	Means that the message type is in binary format
0x00	This field is reserved and can be filled in with zeros
0x02,0x00,0x00,0x00	Means the log is going to be triggered ("ontime")
0x00,0x00,0x00,0x00 0x00,0x00,0x24,0x40	The period of time the log is going to be trigged is 10 seconds
0x00,0x00,0x00,0x00 0x00,0x00,0x00,0x00	Offset time of message when logged
0x00,0x00,0x00,0x00	This log can be removed by "Unlogall" command

Table D.10: Body of "Log rtcaref" Message

Hexadecimal Command	*Description*
0x40,0x00,0x00,0x00	This log is loaded at Com2 of "Base Station" device
0x0A,0x00	Message ID for "rtca1" log
0x00	Means that the message type is in binary format
0x00	This field is reserved and can be filled in with zeros
0x02,0x00,0x00,0x00	Means the log is going to be triggered ("ontime")
0x00,0x00,0x00,0x00 0x00,0x00,0x24,0x40	The period of time the log is going to be trigged is 10 seconds
0x00,0x00,0x00,0x00 0x00,0x00,0x08,0x40	Offset time of message when logged (3 seconds)
0x00,0x00,0x00,0x00	This log can be removed by "Unlogall" command

Table D.11: Body of "Log rtca1" Message

Hexadecimal Command	*Description*
0x40,0x00,0x00,0x00	This log is loaded at Com2 of "Base Station" device
0x5B,0x01	Message ID for "rtcaephem" log
0x00	Means that the message type is in binary format
0x00	This field is reserved and can be filled in with zeros
0x02,0x00,0x00,0x00	Means the log is going to be triggered ("ontime")
0x00,0x00,0x00,0x00 0x00,0x00,0x24,0x40	The period of time the log is going to be trigged is 10 seconds
0x00,0x00,0x00,0x00 0x00,0x00,0x1C,0x40	Offset time of message when logged (7 seconds)
0x00,0x00,0x00,0x00	This log can be removed by "Unlogall" command

Table D.12: Body of "Log rtcaephem" Message

1. The CRC32 checksum of the first message is: **0x2C,0x7A,0xAD,0xBE**.

2. The CRC32 checksum of the second message is: **0x78,0x99,0x30,0x94**.

3. The CRC32 checksum of the third message is: **0x41,0xBE,0x25,0x21**.

4. The CRC32 checksum of the fourth message is: **0xBF,0x43,0x92,0xA0**.

D.3 Rover Station Messages

The following messages need to be written in the same order to configure the rover station:

1. Com com2 57600 n 8 1 n off off

2. Interfacemode com2 rtca none off

The header and body of "Com" message are in Tables D.13 and D.14 respectively:

Hexadecimal Command	*Description*
0xAA	Beginning and present in all Headers
0x44	Beginning and present in all Headers
0x12	Beginning and present in all Headers
0x1C	Total length of Header
0x04,0x00	Command ID for *Com*
0x00	Means the message is going to be in binary format
0xC0	Means the processor is connected to "This port" of the receiver
0x20,0x00	Length of the message not taking into account the header length nor the 4 extra bytes of the CRC32 checksum
0x00,0x00	Sequence number for multiple related *Logs* to be filled in with zeros
0x00	Idle time in the last second between successive logs with same message ID, to be filled in with zeros
0x00	Time status of GPS, because is a request from the processor it can be filled in with zeros
0x00,0x00	GPS week number, not to be considered when requesting a log
0x00,0x00,0x00,0x00	Milliseconds from beginning of GPS week
0x00,0x00,0x00,0x00	Receiver Status ignored when writing data
0x00,0x00	For internal use of the device ignored when writing
0x00,0x00	Software version ignored when writing

Table D.13: Header of "Com" command

Hexadecimal Command	*Description*
0x40,0x00,0x00,0x00	Port of the receiver that is going to be configured This case is "com2"
0x00,0xE1,0x00,0x00	Communication baud rate (bps). This case is 57600bps
0x00,0x00,0x00,0x00	Parity of message. This case it has no parity
0x08,0x00,0x00,0x00	Number of databits. This case it has 8 bits
0x01,0x00,0x00,0x00	Number of stop bits. This case is 1 stop bit
0x00,0x00,0x00,0x00	Handshake. This case it has no handshake
0x00,0x00,0x00,0x00	Echo. 0 value means "off"
0x00,0x00,0x00,0x00	Break. 0 value means "off"

Table D.14: Body of "Com" command

CRC32 checksum: **0xED,0x71,0xBD,0x9D**.

The header and body of the "Interfacemode" command message are presented in Tables D.15 and D.16 respectively:

Hexadecimal Command	*Description*
0xAA	Beginning and present in all Headers
0x44	Beginning and present in all Headers
0x12	Beginning and present in all Headers
0x1C	Total length of Header
0x03,0x00	Command ID for *Interfacemode*
0x00	Means the message is going to be in binary format
0xC0	Means the processor is connected to "This port" of the receiver
0x10,0x00	Length of the message not taking into account the header length nor the 4 extra bytes of the CRC32 checksum
0x00,0x00	Sequence number for multiple related *Logs* to be filled in with zeros
0x00	Idle time in the last second between successive logs with same message ID, to be filled in with zeros
0x00	Time status of GPS, because is a request from the processor it can be filled in with zeros
0x00,0x00	GPS week number, not to be considered when requesting a log
0x00,0x00,0x00,0x00	Milliseconds from beginning of GPS week
0x00,0x00,0x00,0x00	Receiver Status ignored when writing data
0x00,0x00	For internal use of the device ignored when writing
0x00,0x00	Software version ignored when writing

Table D.15: Header of "Interfacemode" command

Hexadecimal Command	*Description*
0x40,0x00,0x00,0x00	Port of the receiver that is going to be configured This case is "com2"
0x03,0x00,0x00,0x00	Reception type. This case is RTCA
0x00,0x00,0x00,0x00	Transmission type. This case is None
0x00,0x00,0x00,0x00	Response Generation. This case it is OFF

Table D.16: Body of "Interfacemode" command

CRC32 checksum: **0x19,0x0A,0x55,0xDA.**

After configuration, any log request can be reported such as:

1. Log bestpos ontime 1.

2. Log bestvel ontime 1.

3. Log rtkpos ontime 1.

4. Log rtkvel ontime 1.

5. Log rtkxyz ontime 1.

All headers are the same as the one shown in Table D.8. The "Log" examples in the list presented request position and velocity. Position is requested by "bestpos", "rtkpos" and "rtkxyz". Velocity is requested by "bestvel", "rtkvel" and "rtkxyz". The "bestpos" and "rtkpos" logs report position vectors in earth's geodetic coordinate frame. The "bestvel" and "rtkvel" logs report velocity vectors in local tangent coordinate frame (a.k.a. geographic frame). The "rtkxyz" log reports position and velocity vectors in ECEF coordinate frame.

D.3.1 "Bestpos" and "Rtkpos" logs

Their response format is similar. The main differences reside in the message ID and that "Bestpos" can compute single point positioning whereas "Rtkpos" can not even be requested unless Base and Rover stations are operating in CDGPS mode. This section is going to explain how to request to the NovAtel device these logs, and later how to interpret the NovAtel device responses.

Request Position

It is assumed Rover and Base are in CDGPS mode. The body of the log message will look like in Table D.17:

Hexadecimal Command	*Description*
0x20,0x00,0x00,0x00	This log is loaded at Com1 of "Rover Station" device
0x2A,0x00/0x8D,0x00	Message ID for "bestpos"/"rtkpos" log
0x00	Means that the message type is in binary format
0x00	This field is reserved and can be filled in with zeros
0x02,0x00,0x00,0x00	Means the log is going to be triggered ("ontime")
0x00,0x00,0x00,0x00 0x00,0x00,0xF0,0x3F	The period of time the log is going to be trigged is 1 second
0x00,0x00,0x00,0x00 0x00,0x00,0x00,0x00	Offset time of message when logged
0x00,0x00,0x00,0x00	This log can be removed by "Unlogall" command

Table D.17: Body of "Log bestpos/rtkpos" Message

- The checksum for "bestpos" command is: **0x66,0xA3,0xBD,0xCC**.
- The checksum for "rtkpos" command is: **0x15,0x7D,0xF5,0x24**

Device Response

The NovAtel will respond with the same format used at the requisition. It will first produce a response that is common to all logs and latter it will respond with the information requested. If the "Log" command was received correctly the "Log" response should look like in Tables D.18 and D.19:

Hexadecimal Command	*Description*
0xAA	Beginning and present in all Headers
0x44	Beginning and present in all Headers
0x12	Beginning and present in all Headers
0x1C	Total length of Header
0x01,0x00	Command ID for *Log*
0x82	Means this is a response message
0x20	Means the device is connected to "com1" of the receiver
0x06,0x00	Length of the message not taking into account the header length nor the 4 extra bytes of the CRC32 checksum
0x00,0x00	Sequence number for multiple related *Logs*
0xFF	Idle time in the last second between successive logs with same message ID
0xB4	Time status of GPS, means it is "finesteering"
0xEE,0x04	GPS week number, not important
0x60,0x5A,0x05,0x13	Milliseconds from beginning of GPS week
0x00,0x00,0x4C,0x00	Receiver Status, Everything OK
0xFF,0xFF	For internal use of the device ignored when reading
0x5A,0x80	Software version ignored when reading

Table D.18: Header of "Log" command response

Hexadecimal Command	*Description*
0x01,0x00,0x00,0x00	Command was received correctly
0x4F,0x4B	Message was received correctly

Table D.19: Body of "Log" Response Message

The CRC32 checksum of this "Log" response is: **0xDA,0x86,0x88,0xEC**. Once the "Log" response is sent by the NovAtel device, the information message (e.g. "Bestpos", "Rtkpos") is sent. Like in all messages, it has a header at the beginning and a checksum at the end.

In order to interpret the message, it is necessary to begin checking fields 4, 5, 8 and 11 of Table D.1. Later check the CRC32 checksum of header and body coincides with the CRC32 checksum emitted by the NovAtel device.

- Field 4 is the header length

- Field 5 is the message ID which in case of "Bestpos" is "0x2A,0x00" and in case of "rtkpos" is "0x8D,0x00"
- Field 8 is the length of the body of the message (72 bytes). This added with the header length and the four bytes of the CRC32 checksum gives the total length of the binary message.
- Field 11 is the time status of the device.

The format of "Bestpos" and "Rtkpos" responses are equal. Table D.20 shows such format. The information was taken from reference [21, Ch. 3.4.3, Ch. 3.4.81]

Field Number	*Field Name*	*Field Type*	*Description*	*Binary Bytes*
1	Header	-	"Bestpos"/"Rtkpos" Header	28
2	Sol_Status	Enum	Solution status	4
3	pos_type	Enum	Position type	4
4	lat	Double	Latitude position	8
5	lon	Double	Longitude position	8
6	hgt	Double	Altitude above mean sea level	8
7	undulation	Float	Relationship between geoid and WGS84 ellipsoid	4
8	datum id#	Enum	Datum ID, lookup table	4
9	lat σ	Float	Latitude standard deviation	4
10	lon σ	Float	Longitude standard deviation	4
11	hgt σ	Float	Height standard deviation	4
12	stn id	Char	Base station ID	4
13	diff_age	Float	Differential age in seconds	4
14	sol_age	Float	Solution age in seconds	4
15	#obs	Uchar	Number of observations tracked	1
16	#GPSL1	Uchar	Number of GPS L1 ranges used in computation	1
17	#L1	Uchar	Number of GPS L1 ranges above the RTK mask angle	1
18	#L2	Uchar	Number of GPS L2 ranges above the RTK mask angle	1
19	-	Uchar	Reserved	1
20	-	Uchar	Reserved	1
21	-	Uchar	Reserved	1
22	-	Uchar	Reserved	1
23	-	Hex	CRC32 checksum	4

Table D.20: Body of Bestpos/Rtkpos Message Responses

D.3.2 "Bestvel" and "Rtkvel" logs

Data format of "Bestvel" and "Rtkvel" are equal though "Bestvel" can compute single point velocity and "Rtkvel" operates only in CDGPS mode.

Request Velocity

The body to request the messages "Bestvel" and "Rtkvel" is shown in Table D.21:

Hexadecimal Command	*Description*
0x20,0x00,0x00,0x00	This log is loaded at Com1 of "Rover Station" device
0x63,0x00/0xD8,0x00	Message ID for "bestvel"/"rtkvel" log
0x00	Means that the message type is in binary format
0x00	This field is reserved and can be filled in with zeros
0x02,0x00,0x00,0x00	Means the log is going to be triggered ("ontime")
0x00,0x00,0x00,0x00 0x00,0x00,0xF0,0x3F	The period of time the log is going to be trigged is 1 second
0x00,0x00,0x00,0x00 0x00,0x00,0x00,0x00	Offset time of message when logged
0x00,0x00,0x00,0x00	This log can be removed by "Unlogall" command

Table D.21: Body of "Log bestvel/rtkvel" Message

- The checksum for "bestvel" command is: **0xD3,0xF1,0xD8,0xAD**.
- The checksum for "rtkvel" command is: **0xEC,0xE2,0xB7,0x55**

Device Response

The device will respond with a "Log" response as explained before. Check fields 4, 5, 8 and 11 of Table D.1. Later check the CRC32 checksum of header and body coincides with the CRC32 checksum emitted by the NovAtel device.

- Field 4 is the header length
- Field 5 is the message ID which in case of "Bestvel" is "0x63,0x00" and in case of "rtkvel" is "0xD8,0x00"
- Field 8 is the length of the body of the message (44 bytes). This added with the header length and the four bytes of the CRC32 checksum gives the total length of the binary message.

- Field 11 is the time status of the device.

The format of "Bestvel" and "Rtkvel" responses are equal. Table D.22 shows such format. The information was taken from reference [21, Ch. 3.4.5, Ch. 3.4.82].

Field Number	*Field Name*	*Field Type*	*Description*	*Binary Bytes*
1	Header	-	"Bestvel"/"Rtkvel" Header	28
2	Sol_Status	Enum	Solution status	4
3	vel_type	Enum	Velocity type	4
4	latency	Float	Latency time of velocity	4
5	age	Float	Differential age in seconds	4
6	hor spd	Double	Horizontal speed [$\frac{m}{s}$]	8
7	trk gnd	Double	Direction of motion over ground with respect to true north measured in degrees	8
8	vert spd	Double	Vertical speed [$\frac{m}{s}$] positive (up) indicates increasing altitude and negative (down) decreasing altitude	8
9	-	Float	Reserved	4
10	-	Float	CRC32 checksum	4

Table D.22: Body of Bestvel/Rtkvel Message Responses

D.3.3 "Rtkxyz" Log

This "log" reports position and velocity information in ECEF coordinates and operates only in CDGPS mode.

Request "Rtkxyz" Log

The body to request the message of "Rtkxyz" is shown in Table D.23:

Hexadecimal Command	*Description*
0x20,0x00,0x00,0x00	This log is loaded at Com1 of "Rover Station" device
0xF4,0x00	Message ID for "rtkxyz" log
0x00	Means that the message type is in binary format
0x00	This field is reserved and can be filled in with zeros
0x02,0x00,0x00,0x00	Means the log is going to be triggered ("ontime")
0x00,0x00,0x00,0x00 0x00,0x00,0xF0,0x3F	The period of time the log is going to be trigged is 1 second
0x00,0x00,0x00,0x00 0x00,0x00,0x00,0x00	Offset time of message when logged
0x00,0x00,0x00,0x00	This log can be removed by "Unlogall" command

Table D.23: Body of "Log rtkxyz" Message

The checksum of "Rtkxyz" request is: **0xD0,0x5D,0x70,0x68**.

Device Response

The device will respond with a "Log" response as explained before. Check fields 4, 5, 8 and 11 of Table D.1. Later check the CRC32 checksum of header and body coincides with the CRC32 checksum emitted by the NovAtel device.

Field Number	*Field Name*	*Field Type*	*Description*	*Binary Bytes*
1	Header	-	"Rtkxyz" Header	28
2	P-sol status	Enum	Solution status	4
3	pos type	Enum	Position type	4
4	P-X	Double	X-coordinate position	8
5	P-Y	Double	Y-coordinate position	8
6	P-Z	Double	Z-coordinate position	8
7	P-X σ	Float	Standard deviation of P-X	4
8	P-Y σ	Float	Standard deviation of P-Y	4
9	P-Z σ	Float	Standard deviation of P-Z	4
10	V-sol status	Enum	Solution status	4
11	vel type	Enum	Velocity type	4
12	V-X	Double	Velocity along X-axis$[\frac{m}{s}]$	8
13	V-Y	Double	Velocity along Y-axis$[\frac{m}{s}]$	8
14	V-Z	Double	Velocity along Z-axis$[\frac{m}{s}]$	8
15	V-X σ	Float	Standard deviation of V-X$[\frac{m}{s}]$	4
16	V-Y σ	Float	Standard deviation of V-Y$[\frac{m}{s}]$	4
17	V-Z σ	Float	Standard deviation of V-Z$[\frac{m}{s}]$	4
18	stn ID	Char	Base station ID	4
19	V-latency	Float	Latency of velocity in seconds	4
20	diff_age	Float	Differential age in seconds	4
21	sol_age	Float	Solution age in seconds	4
22	#obs	Uchar	Number of observations tracked	1
23	#GPSL1	Uchar	Number of GPS L1 ranges used in computation	1
24	#L1	Uchar	Number of GPS L1 ranges above the RTK mask angle	1
25	#L2	Uchar	Number of GPS L2 ranges above the RTK mask angle	1
26	-	Uchar	Reserved	1
27	-	Uchar	Reserved	1
28	-	Uchar	Reserved	1
29	-	Uchar	Reserved	1
30	-	Hex	CRC32 checksum	4

Table D.24: Body of "Rtkxyz" Message Responses

- Field 4 is the header length
- Field 5 is the message ID which in case of "Rtkxyz" is "0xF4,0x00".
- Field 8 is the length of the body of the message (112 bytes). This added with the header length and the four bytes of the CRC32 checksum gives the total length of the binary message.
- Field 11 is the time status of the device.

Table D.24 shows such format. The information was taken from reference [21, Ch. 3.4.83].

D.4 Fix Position Algorithm

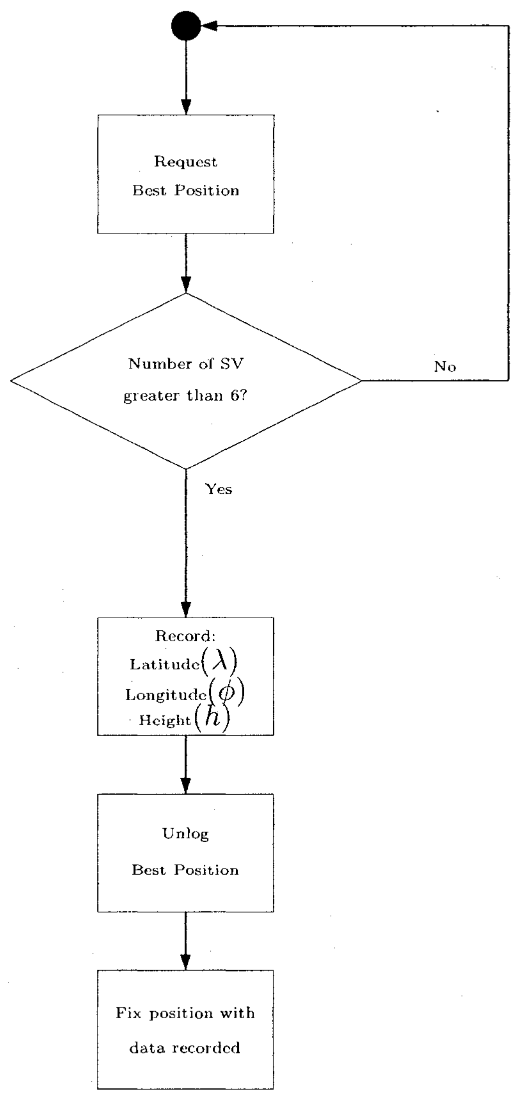


Figure D.1: "Fix position" Algorithm

The application will wait until the antenna has a view of at least six satellites (SVs) in order to fix the base station's position. The log "Bestpos" is synchronous, and if it is left unlogged, unnecessarily it will report the base antenna's position to the base station device. To unlog "Bestpos" one proceeds as follows:

Hexadecimal Command	*Description*
0xAA	Beginning and present in all Headers
0x44	Beginning and present in all Headers
0x12	Beginning and present in all Headers
0x1C	Total length of Header
0x24,0x00	Command ID for *Unlog*
0x00	Means the message is going to be in binary format
0xC0	Means the processor is connected to "This port" of the receiver
0x08,0x00	Length of the message not taking into account the header length nor the 4 extra bytes of the CRC32 checksum
0x00,0x00	Sequence number for multiple related *Logs* to be filled in with zeros
0x00	Idle time in the last second between successive logs with same message ID, to be filled in with zeros
0x00	Time status of GPS, because is a request from the processor it can be filled in with zeros
0x00,0x00	GPS week number, not to be considered when requesting a log
0x00,0x00,0x00,0x00	Milliseconds from beginning of GPS week
0x00,0x00,0x00,0x00	Receiver Status ignored when writing data
0x00,0x00	For internal use of the device ignored when writing
0x00,0x00	Software version ignored when writing

Table D.25: Header of "Unlog" command

Hexadecimal Command	*Description*
0x20,0x00,0x00,0x00	This log is loaded at Com1 of "Base Station" device
0x2A,0x00	Message ID for "Bestpos" log
0x00	Means that the message type is in binary format
0x00	This field is reserved and can be filled in with zeros

Table D.26: Body of "Unlog bestpos" Message

Unlog message CRC32 checksum is **0x9A,0xA8,0xE5,0x7A**. The last step is to fix the position with the position data recorded from the "Bestpos" log with the format explained in this appendix. Latitude, longitude and height data are already given in ieee 754 standard and "Big Endian" format.

Appendix E

Sensor Calibration

The calibration process is done off-line. It is assumed that the GPS system is unbiased and its measurements will only be corrupted by white noise thus no calibration is needed. However, the magnetometers, accelerometers and gyroscopes sensors of the IMU need to be calibrated in order to obtain a measurement as unbiased as possible and to be able to use equation (5.71) in the Kalman Filter algorithm.

The main tasks of the calibration process are: (1) To determine all sensor DC offset bias parameters (a.k.a. "turn-on to turn-off" or "null shift" in any sensor datasheet [10, Ch. 3.2]) and (2) To find all sensor variances in order to determine the input covariance "Q" and the output covariance "R" matrices. As it should be well known by now, there are three vectorial sensors incorporated in the IMU which are accelerometers, gyroscopes and magnetometers. Calibration of accelerometer and gyroscope sensors is much more straightforward than magnetometer calibration and therefore they will be explained first.

E.1 Accelerometer and Gyroscope Calibration

Let $s_m(t)$ be the signal measured by a sensor. Let $s_t(t)$ be its true signal. Let $b(t)$ be the noise component of the true signal. A simplified model of the measured signal in terms of the true signal is:

$$s_m(t) = s_t(t) + b(t) \tag{E.1}$$

Where the noise component $b(t)$ is modeled as in reference [10, Ch. 3.2]:

$$b(t) = b_{DC} + b_1(t) + \nu(t) \tag{E.2}$$

$b_1(t)$ represents the random walk component and $\nu(t)$ represents the white noise component of either accelerometer or gyroscope sensor.

The model for the random walk component is described by the following equation:

$$\dot{b}_1(t) = \omega(t) \tag{E.3}$$

Where $\omega(t)$ is a white noise process. The variance of the process noise $\omega(t)$ is usually much smaller than the variance of $\nu(t)$. However, the variance of a random walk process is unbounded and grows as time passes ($\sigma_{b_1}^2 = \sigma_\omega^2 t$). The model in equation (E.3) could be physically unrealistic because it has an ever increasing variance[10, Ch. 3.3.1]. Gauss-Markov process, is another way to model the time varying bias component $b_1(t)$ using the following equation:

$$\dot{b}_1(t) = -\frac{1}{\tau} b_1(t) + \omega(t) \tag{E.4}$$

Both models seem to each other, in fact the Gauss-Markov process approaches the random walk in the limit as τ approaches infinity. Allan variance chart might be used to determine roughly at what time will the $b_1(t)$ dominate over $\nu(t)$[10, Ch. 3.3.1]. Let that time be t_0.

Zero input response data collection (i.e. UAV is static) with the vehicle leveled and aligned to true north is necessary to determine the DC offset bias (i.e. b_{DC}) of accelerometers and gyroscopes. The time average of the measurements should be taken until it reaches the time determined at the allan variance chart (t_0).

For the case of accelerometers equation (E.5) is used:

$$b_{a_{DC}}^b(kT_{ins}) = \left(1 - \frac{1}{k}\right) b_{a_{DC}}^b((k-1)T_{ins}) + \frac{1}{k}\left(\tilde{f}^b(kT_{ins}) - g^n\right) \tag{E.5}$$

and for the case of gyroscopes equation (E.6) is used:

$$b^b_{\omega_{DC}}(kT_{ins}) = \left(1 - \frac{1}{k}\right) b^b_{\omega_{DC}}((k-1)T_{ins}) + \frac{1}{k}\tilde{\omega}^b_{ib}(kT_{ins}) \tag{E.6}$$

It should be noted that the gravity vector g^n was subtracted from the accelerometer measurement $\tilde{f}^b$ because the IMU devices measure specific force. The vehicle is leveled and stationary therefore the specific force measured by the IMU sensor is the gravity vector (i.e. $f^b = g^n$).

The advantage of using recursive equations relies on the fact that less computational memory is required than computing the average of the batched data. Any initial condition could be used for $b^b_{a_{DC}}(0)$ and $b^b_{\omega_{DC}}(0)$. The solution will converge to the average of the data collected [4, Ch. 4.2] and will be an unbiased estimate of the DC offset of gyroscope and accelerometer sensors.

The other main task in the calibration process of accelerometer and gyroscope sensors is to find the variance of the band limited white noise $\nu(t)$. There are two methods of finding $\nu(t)$. Both methods use the data collected when the UAV is static, leveled and ideally aligned with true north.

E.1.1 Standard Deviation Method

The first method consists in removing the DC offset bias found in equations (E.5) and (E.6) for accelerometers and gyroscopes respectively. Then the square of the standard deviation computed for the data collected from time zero to t_0 is an unbiased estimate of the variance of $\nu(t)$.

To compute the standard deviation of accelerometers use the following equation:

$$\sigma_{f^b} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left((\tilde{f}^b - g^n)_i - \overline{(\tilde{f}^b - g^n)}\right)^2} \tag{E.7}$$

To compute the standard deviation of gyroscopes use the following equation:

$$\sigma_{\omega_{ib}^{b}} = \sqrt{\frac{1}{N-1}\sum_{k=1}^{N}\left((\tilde{\omega}_{ib}^{b})_k - \overline{(\tilde{\omega}_{ib}^{b})}\right)^2} \tag{E.8}$$

The variance found using equations (E.7) and (E.8) assumes the data is previously taken and therefore the process is done offline.

E.1.2 Allan Variance Method

Viewed as the time domain equivalent of power spectrum density. It gives power as a function of averaging time [10, App. C]. From the collected data set, and preferably but not necessary removing the DC offset values of the sensors, define a vector of time averages $\tau = [\tau_1, \tau_2, \ldots, \tau_N]^T$.

Each of these time averages divide the total time "T" of data collection into M_j subintervals $\left(M_j = \frac{T}{\tau_j}\right)$. Where $j = 1, \ldots, N$. Compute the mean value of each subinterval "i" of each time average τ_j, where $i = 1, \ldots, M_j$. Let each mean value be $\overline{x}_{j,i}$.

The Allan variance is computed as follows:

$$\sigma^2(\tau_j) = \frac{1}{2(M_j - 1)}\sum_{i=1}^{M_j-1}\left(\overline{x}_{j,i+1} - \overline{x}_{j,i}\right)^2 \tag{E.9}$$

On a log-log scale, plot the allan variance $\sigma(\tau_j)$ of each time average against its average time. It is recommended to use more than nine time averages τ_j, also that every time average should be less or equal to half the total time of data collection $\left(i.e.\, \tau_j < \frac{T}{2}\ \ \forall j\right)$.

Table C.1 of reference [10, App. C] provide typical noise process models and how they should look like in an Allan Variance Chart. Figure E.1 displays the allan variance charts of white noise, random walk and gauss-markov noise processes.

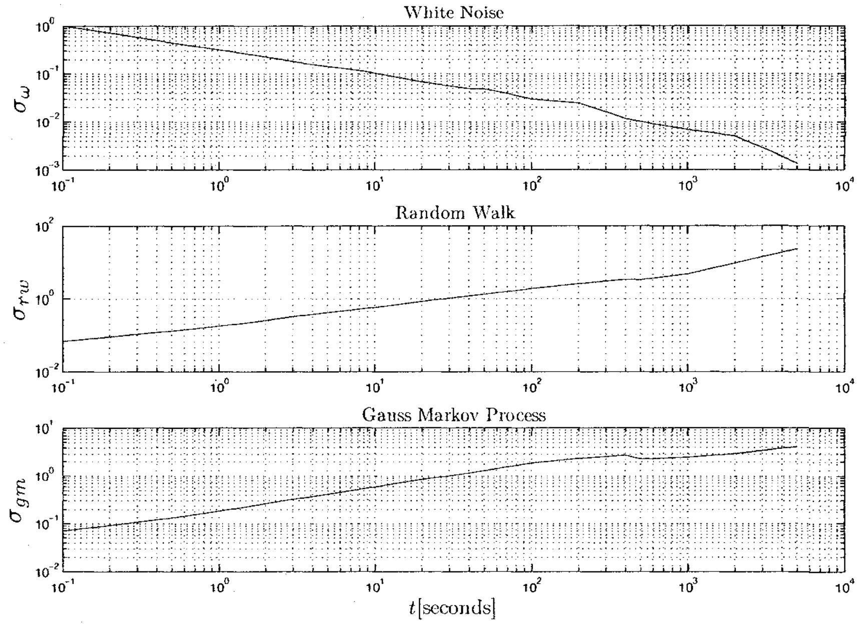


Figure E.1: Allan Variance Charts

It is assumed that other source of noise aside of white noise affects the measurement significantly several seconds after. In a sensor there is a combination of errors. Usually the sensor measurement is modeled as the true value corrupted by white noise and random walk noise or white noise and Gauss-Markov process noise. To evaluate the white noise variance of a sensor it suffices to look at the Allan variance chart for the value of the standard deviation when the average time is one second (*i.e.* $\sigma(1)$) and later scaling this value properly. The scaled value of the standard deviation found is also known in sensor datasheets as the "*angle random walk*" because it is integrating the gyro measurement in time in order to determine the attitude of the sensor.

It is interesting to see why at an average time of one, the allan variance defines the variance of the noise disturbance in the sensor measurement. The relation between the Allan variance and the Power Spectral Density (PSD) is [10, App. C.3]:

$$\sigma^2 \left(\tau_{av}\right) = \frac{4}{\pi \tau_{av}} \int_0^\infty S \left(\frac{u}{\pi \tau_{av}}\right) \frac{\sin^4 u}{u^2} \, du \tag{E.10}$$

Variance of White Noise

The PSD of a white noise process is a flat line with an amplitude equal to the noise power. Let that amplitude be the variance of the white noise σ_ω^2 then the PSD will be $S(j\Omega) = \sigma_\omega^2\ \forall\Omega$. Using equation (E.10):

$$\sigma^2(\tau_{av}) = \frac{4\sigma_\omega^2}{\pi\tau_{av}} \int_0^\infty \frac{\sin^4 u}{u^2}\, du = \frac{4\sigma_\omega^2}{\pi\tau_{av}} \frac{\pi}{4} = \frac{\sigma_\omega^2}{\tau_{av}} \tag{E.11}$$

The variance of the white noise process is therefore computed when $\tau_{av} = 1$. However, this variance must be properly scaled by the sampling time of the sensor. In case of white noise corruption, the scaling to the standard deviation of $\sigma(\tau_{av} = 1)$ is $\frac{1}{\sqrt{T}}$. For example looking at the top plot of figure E.1, which is a typical representation of a measurement corrupted by white noise only, the value of the variance at an average time of one second is $\sigma(\tau_{av} = 1) = 0.3146$. The sampling time of the band limited white noise is $T = 0.1s$. The standard deviation of the white noise process is therefore $\sigma_\omega = \frac{0.3146}{\sqrt{0.1}} = 1$.

Variance of Random Walk Noise

A random walk process as explained before, is obtained by integrating white noise. Its PSD is therefore $S(j\Omega) = \frac{\sigma_\omega^2}{\Omega^2}$. Using equation (E.10):

$$\sigma^2(\tau_{av}) = \frac{4\sigma_\omega^2}{\pi\tau_{av}} \int_0^\infty \frac{Q^2}{\left(2\pi\frac{u}{\pi T}\right)^2} \frac{\sin^4 u}{u^2}\, du = \frac{Q^2\tau_{av}}{3} \tag{E.12}$$

The scaling factor to the standard deviation of $\sigma(\tau_{av} = 1)$ is $\sqrt{\frac{3}{T}}$. For example looking at the middle plot of figure E.1, which is a typical representation of a measurement corrupted by random walk noise only, the value of the variance at an average time of one second is $\sigma(\tau_{av} = 1) = 0.1828$. The sampling time of the band limited white noise is $T = 0.1s$. The standard deviation of the white noise that produces the random walk process is therefore $\sigma_\omega = 0.1828 * \sqrt{\frac{3}{0.1}} = 1$.

Variance of Gauss-Markov Noise

The PSD of a Gauss-Markov process is:

$$S(j\Omega) = \frac{\sigma_\omega^2 \alpha^2}{1 + (\alpha\Omega)^2} \tag{E.13}$$

There is no close expression for the scaling factor to the standard deviation of $\sigma(\tau_{av} = 1)$ with a Gauss-Markov process noise. The integral in equation (E.10) must be solved numerically. For example for $\alpha = 1000s$ and $T = 0.1s$, the integral value is $1.96e - 7$ and equation (E.10) reduces to:

$$\sigma(\tau_{av} = 1) = 1.96e - 7\frac{400000\sigma_\omega^2}{\pi} \tag{E.14}$$

Looking at the bottom plot of figure E.1, the value of the variance at an average time of one second is $\sigma(\tau_{av} = 1) = 0.1828$. The sampling time of the band limited white noise is $T = 0.1s$. The standard deviation of the white noise that produces the gauss-markov process is therefore:

$$\sigma_\omega = 0.1828 * \sqrt{\frac{\pi}{4e5 \times 1.96e - 7}} \approx 1 \tag{E.15}$$

E.2 Magnetometer Calibration

A similar procedure as with the gyroscopes and accelerometers is followed for magnetometer calibration. Determining the DC offset bias as well as the white noise covariance and correlated noise models for the magnetometer sensor follows a similar procedure as explained in the last section. However, these sensors have other type of disturbances and a stochastic model won't suffix for calibration purposes because they are deterministic perturbations. Furthermore initial alignment is necessary in order to determine the DC offset bias component accurately.

E.2.1 Alignment Argument

Similar to the accelerometer sensors, the magnetic field vector (h^n) is different than zero. It is necessary to subtract the magnetic field vector from the sensor measurements in order to determine accurately its bias model. Contrary to the accelerometer sensor case, when decomposing the earth magnetic field vector in a local geographic frame, all three components of the frame will be involved. As a consequence, in order to determine the DC offset bias of the magnetometers accurately, the IMU must be aligned and leveled.

Let's assume that IMU axes are parallel to UAV's. The magnetic field measured by the magnetometers assuming zero external disturbances will be the earth's magnetic field decomposed in its sensitive axes components. Assume also these axes represent the body frame coordinate reference. At $53^{0}31'37''$ latitude north, $113^{0}31'37''$ longitude west and $707m$ of altitude, it can be verified that the magnetic field vector is $h^n = [0.138849; 0.04076; 0.56231]$ $Gauss$ with any online magnetic field calculator such as in [22]. Appendix C explains in detail how to compute such vector.

The magnetic field vector is known when is decomposed in navigation frame coordinates (i.e. geographic frame), however the attitude of the vehicle is unknown. The magnetic field vector sensed by the IMU sensor ($\tilde{h}^b$) and the earth's magnetic field vector (h^n) must match, thus a Rotation of the measured $\tilde{h}^b$ by a Direction Cosine Matrix (DCM) is needed. Wabah's vector matching problem can be used in the determination of the vehicle's attitude along with the measured specific force ($i.e.$ $\tilde{f}^b$) and the cross product of specific force and magnetic field ($i.e.$ $\tilde{f}^b \times \tilde{h}^b$). In reference [23] it is stated that the Wabah's attitude determination problem is:

$$L(A) = \frac{1}{2}\sum_{i=1}^{N}\left(v_i - C_n^b u_i\right) \tag{E.16}$$

Where u_i is the set vectors decomposed in geographic frame coordinates which are theoretically known. v_i is the set of vectors that are measured by the IMU sensor and thus decomposed in body frame coordinates. Note that $\left[\tilde{f}^b, \tilde{h}^b, \tilde{f}^b \times \tilde{h}^b\right]_i \in \{v_i\}$, $[f^n, h^n, f^n \times h^n]_i \in \{u_i\}$, $C_n^b\,[f^n, h^n, f^n \times h^n]_i = \left[\tilde{f}^b, \tilde{h}^b, \tilde{f}^b \times \tilde{h}^b\right]_i$ and h^n, f^n and $f^n \times h^n$ are the earth magnetic field vector, the gravity field vector and the cross product of gravity field and magnetic field vectors respectively which are known. Let:

$$\begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} u_1 & \vdots & u_2 & \vdots & \dots & \vdots & u_N \end{bmatrix} \\ \begin{bmatrix} v_1 & \vdots & v_2 & \vdots & \dots & \vdots & v_N \end{bmatrix} \end{bmatrix} \tag{E.17}$$

The Optimal solution to (E.16) is[23]:

$$C_n^b = VR^{-1}U^T \left(UR^{-1}U^T\right)^{-1} \tag{E.18}$$

In case the vectors $\left[f^b,\, h^b,\, f^b \times h^b\right]_i$ are averaged then the DCM matrix will be[4, Ch. 6.8.8.1]:

$$C_b^n = \left[f^n \;\vdots\; h^n \;\vdots\; f^n \times h^n\right] \left[\tilde{f}^b \;\vdots\; \tilde{h}^b \;\vdots\; \tilde{f}^b \times \tilde{h}^b\right]^{-1} \tag{E.19}$$

Note that $C_n^b = C_b^{n^T}$. The matrices found in equations (E.18) and (E.19) need to be normalized. Furthermore it assumes unbiased measurements. What can be done at most is to have unbiased measurements of $\tilde{f}^b$ because the magnetic field measurement have not been calibrated yet. Thus the DCM will not be a good estimate in order to determine the DC offset bias of the magnetometers.

The vehicle should be leveled and aligned because the IMU sensor measures magnetic field decomposed in body frame coordinates and any change in the UAV's orientation will be seen as a rotation of the magnetic field vector by the IMU sensor unless all its Euler angles are zero (*i.e.* $\phi = 0,\, \theta = 0,\, \psi = 0$).

E.2.2 Alignment Procedure

A flat surface guarantees that the roll and pitch of the vehicle are zero (*i.e.*$\phi = 0,\, \theta = 0$). The GPS system has the option of reporting the heading of its antenna with respect to true north. Alignment is the process of setting the heading of the vehicle at zero degrees with respect to true north.

The GPS system consist of two antennas and two processors. For in depth instructions on how to use the system, the interested reader is encouraged to check references [18] and [21]. The objective is to set up the two antennas to operate in carrier-phase differential mode (CDGPS) improving the system's accuracy. In order to do so, it is necessary to divide the system into base and rover station.

Base Station

The base station consists of an antenna and a processor unit. It is possible to communicate with the processor unit if properly formatted messages are sent to it. The base station antenna position must be determined.

If a survey of the geodetic coordinates of the specific place where the UAV is going to fly was done before, its results can be used to fix the base station antenna position. If there is no knowledge of the coordinates, they can be requested to the base station processor unit using the "Bestpos" command.

After position is fixed, the base station should report: (1)satellite observation information, (2)position information, (3)computed corrections and (4)raw satellite ephemeris information with the "Rtcaobs" command every second, the "Rtcaref" command every ten seconds, the "Rtca1" command every ten seconds and "Rtcaephem" command every ten seconds respectively[21]. Appendix D explains in detail the main commands used to communicate with the Novatel processor devices.

Rover Station

The rover station is the UAV. The other GPS antenna and its processor are going to be mounted on it. After proper configuration of the base station, the rover can request any position and/or velocity information.

In order to take advantage of the CDGPS precision, the commands "Rtkpos", "Rtkvel" and "Rtkxyz" are used to report position and velocity in different coordinate frames. For alignment purposes the "Rtkvel" command is the most important because it reports the actual direction of motion of the antenna over ground with respect to true north in degrees[18]. This is the heading angle ψ. The UAV must be rotated while in CDGPS mode until the heading angle report is zero. The flat surface in conjunction with the heading angle equal to zero guarantees the required leveling and alignment for sensor calibration.

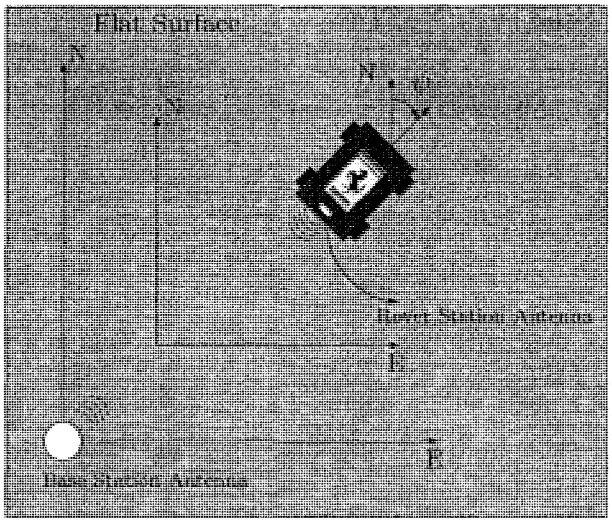


Figure E.2: Alignment Process

E.2.3 Hard Iron Compensation

Besides the stochastic errors presented in the magnetometers, there are deterministic errors that will bias the measurements of these sensors. Hard Iron errors are common in magnetometers and are the most predominant source of interference.

Hard Iron errors consist on the superposition of any time invariant magnetic field to the earth's magnetic field in the measurements. Any material can cause interference if magnetized, but for hard iron interference to occur, the magnetized object must keep the same distance and attitude towards the magnetometer sensors and have a time invariant magnetic field source. Several examples for hard iron interference present in UAV exist such as current carrying wires and permanently magnetized thin metals fixed in the UAV.

The locus of outputs for an error free magnetometer triad set is a sphere centered at the origin. The radius of the sphere will be equal to the strength of the earth's magnetic field vector[10, Ch. 3.11.3]. The measurement model of the magnetometers considering hard iron interference only is:

$$\tilde{h}^b = h^b + \delta h^b \tag{E.20}$$

In a local area, the magnetic field magnitude is constant. It will vary slightly if long distances are covered. Thus the sum of the squares of the components of the magnetic field will result in a shifted-center sphere described by the following equation[10, Ch. 3.11.3]:

$$|h^n|^2 = \left(h_u^b - \delta h_{u0}^b\right)^2 + \left(h_v^b - \delta h_{v0}^b\right)^2 + \left(h_{w0}^b - \delta h_w^b\right)^2 \tag{E.21}$$

Least square algorithm of the linearized model around the measurement bias due to hard iron effects (*i.e.* δh^b) will estimate the center of the sphere. Detailed explanation is given in [10, Ch. 3.11.3].

In [7] Hard iron compensation is made by spinning the UAV around its three orthogonal axes with the IMU sensor strapped down into it. In each spin, a software collects data. When the data collection is ready, for each spin it finds the minimum and maximum values. A minimum will occur when the earth's magnetic field component is off-phased by 180^0 from the sensitive axis of the magnetometer and a maximum when they are in phase. The average of minimum and maximum values without hard iron interference should be zero. Each average will represent a coordinate of the center of the sphere.

E.2.4 Soft Iron Compensation

Soft iron interference consists in the generation and superposition of an artificial magnetic field to the earth's magnetic field due to materials that generate their own magnetic field in response to an external source of magnetic field. The most common external source is the Earth's magnetic field. Depending on the UAV's attitude, the ferromagnetic material will generate a magnetic field in response to the earth's source. The measurement model of the magnetometers including soft iron interference only is[10, Ch. 3.6.2]:

$$\begin{bmatrix} \tilde{h}_u^b \\ \tilde{h}_v^b \\ \tilde{h}_w^b \end{bmatrix} = \begin{bmatrix} \alpha_{xx} & \alpha_{xy} & \alpha_{xz} \\ \alpha_{yx} & \alpha_{yy} & \alpha_{yz} \\ \alpha_{zx} & \alpha_{zy} & \alpha_{zz} \end{bmatrix} \begin{bmatrix} h_u^b \\ h_v^b \\ h_w^b \end{bmatrix} \tag{E.22}$$

The terms $\alpha_{xx}, \alpha_{xy}, ..., \alpha_{zz}$ represent the effective soft iron coefficients. They are the constants of proportionality between the applied magnetic field and the induced magnetic field in the material [10, Ch. 3.11.1]. The effect that soft iron interference has on the locus of the measurement is to:

1. Re-shape locus into an ellipsoid instead of an spheroid.

2. Change the orientation of the ellipsoid.

It is easier to see the effect in two dimensions because there is only one degree of freedom and thus one angle, however in three dimensions there are three degrees of freedom and three euler angles should be used to express the rotation of the locus (i.e. ellipsoid) of the magnetometer measurements due to soft iron interference.

E.3 Sensor Parameter Results

Data of specific force, rotation rate and magnetic field vectors was collected from the IMU sensor for 1800s. The main tasks were: (1)To find the DC offset bias of the vector measurements to each axis and (2)To determine the input covariance and output covariance matrices "Q" and "R" respectively. The nine DC offset biases are easily determined during warm up while the vehicle is leveled and aligned. Matrices "Q"(??) and "R"(??) given in section 5.2.2 were approximations that assumed equal parameters for the three orthogonal axes of each vector measurement, therefore they must be determined.

The methods used in the determination of the covariance matrices were the Allan Variance and the signals auto-covariances. There were seventeen allan averaging times. Quantization, white, flickering, random walk and rate ramp are well known modeled sources of noise that corrupts a sensor measurement. The allan variance as a function of averaging time is given by[24]:

$$\sigma^2(\tau) = \frac{3Q^2}{\tau^2} + \frac{N^2}{\tau} + \frac{2B^2\ln(2)}{\pi} + \frac{K^2\tau}{3} + \frac{R^2\tau^2}{2} = \sum_{n=-2}^{2} C_n\tau^n \qquad \text{(E.23)}$$

Least square estimation (LSE) method is used in conjunction with the allan variance plots to asses noise in the measurements and determine the noise constants Q, N, B, K and R.

The LSE problem will minimize the square error given by the difference between the allan variances [*i.e.* $\sigma^2(\tau_1), \sigma^2(\tau_2), \ldots, \sigma^2(\tau_17)$] computed from the data taken and the allan variances computed from equation (E.23) as follows:

$$X = \begin{bmatrix} \tau_1^{-2} & \tau_1^{-1} & 1 & \tau_1 & \tau_1^2 \\ \tau_2^{-2} & \tau_2^{-1} & 1 & \tau_2 & \tau_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \tau_{17}^{-2} & \tau_{17}^{-1} & 1 & \tau_{17} & \tau_{17}^2 \end{bmatrix}$$

$$\theta = \begin{bmatrix} C_{-2} \\ C_{-1} \\ C_0 \\ C_1 \\ C_2 \end{bmatrix}$$

$$Y = \begin{bmatrix} \sigma^2(\tau_1) \\ \sigma^2(\tau_2) \\ \vdots \\ \sigma^2(\tau_{17}) \end{bmatrix}$$

$$\min \left(Y^T - \theta^T X^T\right)(Y - X\theta) \tag{E.24}$$

The optimal solution to (E.24) is $\theta = \left(X^T X\right)^{-1} X^T Y$. Noise coefficients Q, N, B, K and R are then easily solved equating the coefficients in (E.23).

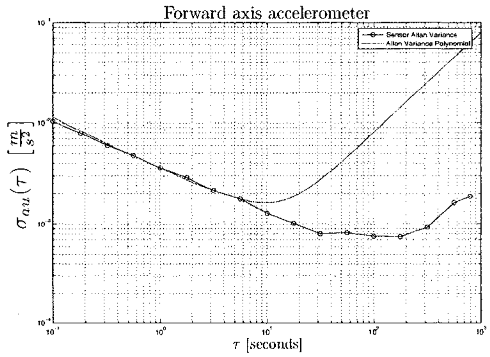
E.3.1 Accelerometers

Figure E.3 shows the allan variance and auto-covariance plots of specific force vector data.

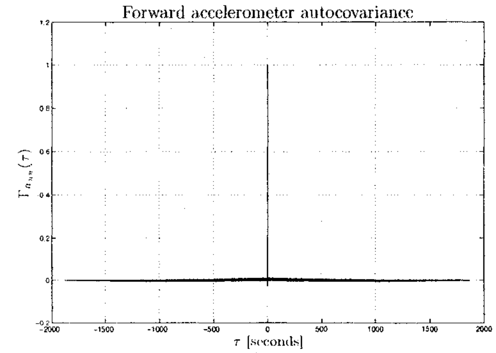
Forward Axis Noise Model

Figure E.3(a) shows roughly that during the first tenths of seconds, the acceleration measurement is corrupted mostly by white noise ($-\frac{1}{2}$ slope until $\tau \approx 20s$). After twenty seconds, the measurement is corrupted mostly by flickering noise. After two hundred seconds the measurement starts being corrupted by random walk process noise.

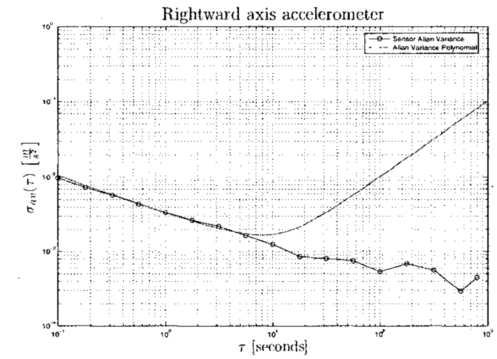
Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:



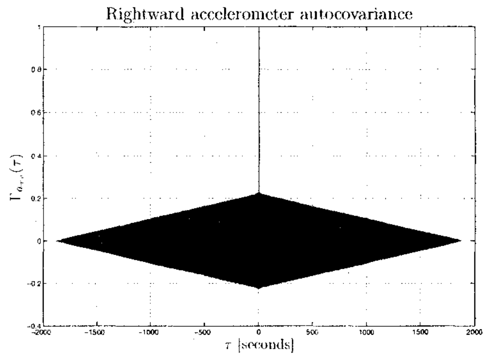
(a) Forward Allan Variance $\sigma_{au}(\tau)$



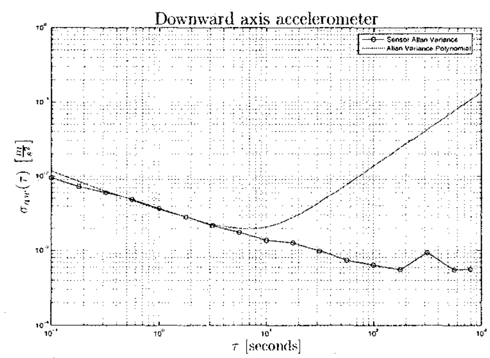
(b) Forward Auto-Covariance $\Gamma_{a_{uu}}(\tau)$



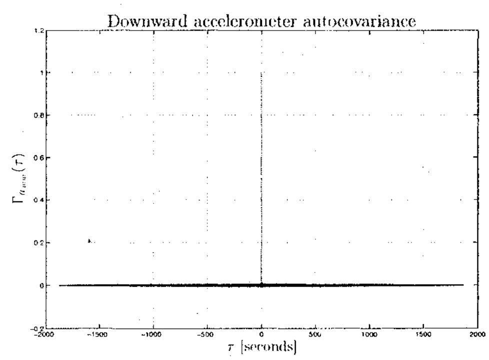
(c) Rightward Allan Variance σ_{av}



(d) Rightward Auto-Covariance $\Gamma_{a_{vv}}$



(e) Downward Allan Variance σ_{aw}



(f) Downward Auto-Covariance $\Gamma_{a_{ww}}$

Figure E.3: Accelerometer Covariance Determination

$Parameter$	$Q_{au}\left[\frac{m}{s}\right]$	$N_{au}\left[\frac{\frac{m}{s}}{\sqrt{s}}\right]$	$B_{au}\left[\frac{m}{s^2}\right]$	$K_{au}\left[\frac{\frac{m}{s^2}}{\sqrt{s}}\right]$	$R_{au}\left[\frac{m}{s^3}\right]$
$Value$	1.94e-4	0.0034	0.0016	7.98e-5	2.9e-4

Table E.1: Forward Accelerometer Noise Coefficients

The sampling frequency of the forward accelerometer is 100Hz (i.e. T=0.01s), therefore the white noise standard deviation is given by:

$$\sigma_{au} = \frac{N_{au}}{\sqrt{T}} = \frac{0.0034}{\sqrt{0.01}} = 0.034\left[\frac{m}{s^2}\right] \tag{E.25}$$

To determine the random walk standard deviation (which is needed for matrix "Q"), the following equation is used:

$$\sigma_{b_{au}} = K_{au}\sqrt{\frac{3}{T}} = 7.98e-5\sqrt{\frac{3}{0.01}} = 0.0014\left[\frac{m}{s^3}\right] \tag{E.26}$$

The auto-covariance plot of the forward axis acceleration signal displayed in figure E.3(b), shows that the measurement is mainly corrupted by white noise because it looks like a sharp impulse. Therefore, it will be assumed that for the output covariance matrix "R" there will be no source of noise other than white noise.

Rightward Axis Noise Model

Figure E.3(c) shows roughly that white noise is the primary source of disturbance in the measurement of acceleration during the first twenty seconds. After twenty seconds it is impossible to infer from the plot what types of noises are affecting the measurements.

Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:

$Parameter$	$Q_{av}\left[\frac{m}{s}\right]$	$N_{av}\left[\frac{\frac{m}{s}}{\sqrt{s}}\right]$	$B_{av}\left[\frac{m}{s^2}\right]$	$K_{av}\left[\frac{\frac{m}{s^2}}{\sqrt{s}}\right]$	$R_{av}\left[\frac{m}{s^3}\right]$
$Value$	1.76e-4	0.0032	0.0014	1.02e-4	2.62e-6

Table E.2: Rightward Accelerometer Noise Coefficients

The auto-covariance plot of the rightward axis acceleration signal displayed in figure E.3(d), shows that the measurement is corrupted by other sources besides white noise because it does not look like a sharp impulse but like a decaying function.

For state update (i.e. Matrix "Q") a random walk noise disturbance is assumed. For measurement update (i.e. Matrix "R") a Gauss-Markov noise process is assumed. The reason for this is that the random walk noise process is modeled in the bias error state ($i.e.\ \dot{b} = \omega$) but since the EKF uses the same set of accelerometers as the aiding measurement, the output covariance would look like $R = K^2 t$ which is unstable. With a Gauss-Markov process noise assumption, the output covariance will be $R = \sigma_{GM}^2 = \frac{K^2 \tau}{2}$ [9] which is stable(τ is the correlation time). Furthermore, a random walk is "possibly a limiting case of an exponentially correlated noise with long correlation time" [25, Ch. 5.2.1].

The white noise standard deviation is given by:

$$\sigma_{av} = \frac{N_{av}}{\sqrt{T}} = \frac{0.0032}{\sqrt{0.01}} = 0.032 \left[\frac{m}{s^2}\right] \tag{E.27}$$

To determine the random walk standard deviation (which is needed for matrix "Q"), the following equation is used:

$$\sigma_{b_{av}} = K_{av}\sqrt{\frac{3}{T}} = 1.02e-4\sqrt{\frac{3}{0.01}} = 0.0018 \left[\frac{m}{s^3}\right] \tag{E.28}$$

The Gauss-Markov process is characterized by an exponentially decaying autocorrelation function as shown in equation (E.29). Thus it is necessary to fit the plotted autocovariance given in figure E.3(d) with the exponential autocorrelation.

$$R_{xx}(\tau) = \sigma_{GM_{av}}^2 e^{-\frac{1}{\alpha}|\tau|} \tag{E.29}$$

The autocovariance of the acceleration measurement is normalized so that $\Gamma_{a_{vv}}(\tau = 0) = 1$. A sharp impulse is seen in figure E.3(d) at $\tau = 0$. However, the curve starts decaying from $\Gamma_{a_{vv}}(\tau = 0) = 0.24$. Therefore, in order to find the autocorrelation time constant α, it suffices to find the time at which the magnitude of the autocovariance of the signal reaches $\Gamma_{a_{vv}}(\tau) = \frac{0.24}{e}$ which is $\alpha = 1071.7s$.

To determine the Gauss-Markov standard deviation (needed for matrix "R"), the following equation is used:

$$\sigma_{GM_{av}} = K_{av}\sqrt{\frac{\alpha}{2}} = 1.02e-4\sqrt{\frac{1071.7}{2}} = 0.0024\left[\frac{m}{s^2}\right] \tag{E.30}$$

Downward Axis Noise Model

Figure E.3(c) shows roughly that white noise is the primary source of disturbance in the measurement of acceleration during the first twenty seconds. After twenty seconds it is impossible to infer from the plot what types of noises are affecting the measurements.

Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:

Parameter	$Q_{aw}\left[\frac{m}{s}\right]$	$N_{aw}\left[\frac{\frac{m}{s}}{\sqrt{s}}\right]$	$B_{aw}\left[\frac{m}{s^2}\right]$	$K_{aw}\left[\frac{\frac{m}{s^2}}{\sqrt{s}}\right]$	$R_{aw}\left[\frac{m}{s^3}\right]$
Value	2.9e-4	0.0034	0.0018	1.35e-4	3.51e-6

Table E.3: Downward Accelerometer Noise Coefficients

The white noise standard deviation is given by:

$$\sigma_{aw} = \frac{N_{aw}}{\sqrt{T}} = \frac{0.0034}{\sqrt{0.01}} = 0.034\left[\frac{m}{s^2}\right] \tag{E.31}$$

To determine the random walk standard deviation (which is needed for matrix "Q"), the following equation is used:

$$\sigma_{b_{aw}} = K_{aw}\sqrt{\frac{3}{T}} = 1.35e-4\sqrt{\frac{3}{0.01}} = 0.0023\left[\frac{m}{s^3}\right] \tag{E.32}$$

The auto-covariance plot of the forward axis acceleration signal displayed in figure E.3(f), shows that the measurement is mainly corrupted by white noise because it looks like a sharp impulse. Therefore, it will be assumed that for the output covariance matrix "R" there will be no source of noise other than white noise.

Input and Output Covariance Submatrices

The input covariance matrices due to accelerometer measurements are:

$$Q_{a\omega} = \begin{bmatrix} \sigma_{au}^2 & 0 & 0 \\ 0 & \sigma_{av}^2 & 0 \\ 0 & 0 & \sigma_{aw}^2 \end{bmatrix} = \begin{bmatrix} 0.034^2 & 0 & 0 \\ 0 & 0.032^2 & 0 \\ 0 & 0 & 0.034^2 \end{bmatrix} \tag{E.33}$$

$$Q_{ab_\omega} = \begin{bmatrix} \sigma_{b_{au}}^2 & 0 & 0 \\ 0 & \sigma_{b_{av}}^2 & 0 \\ 0 & 0 & \sigma_{b_{aw}}^2 \end{bmatrix} = \begin{bmatrix} 0.0014^2 & 0 & 0 \\ 0 & 0.0018^2 & 0 \\ 0 & 0 & 0.0023^2 \end{bmatrix} \tag{E.34}$$

The output covariance matrix due to accelerometer measurements is:

$$R_a = \begin{bmatrix} \sigma_{au}^2 & 0 & 0 \\ 0 & \sigma_{av}^2 + \sigma_{GM}^2 & 0 \\ 0 & 0 & \sigma_{aw}^2 \end{bmatrix} = \begin{bmatrix} 0.034^2 & 0 & 0 \\ 0 & 0.032^2 + 0.0024^2 & 0 \\ 0 & 0 & 0.034^2 \end{bmatrix} \tag{E.35}$$

E.3.2 Gyroscopes

Figure E.4 shows the allan variance and auto-covariance plots of specific force vector data.

Forward Axis Noise Model

The allan variance plot displayed in figure E.4(a) shows roughly that after twenty seconds the flickering noise starts corrupting the rotation measurement. Before that, white noise is the dominant source of disturbance.

Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:

$Parameter$	$Q_{\omega u}\,[rad]$	$N_{\omega u}\left[\frac{rad}{\sqrt{s}}\right]$	$B_{\omega u}\left[\frac{rad}{s}\right]$	$K_{\omega u}\left[\frac{\frac{rad}{s}}{\sqrt{s}}\right]$	$R_{\omega u}\left[\frac{rad}{s^2}\right]$
$Value$	3.47e-5	7.34e-4	2.07e-4	1.06e-5	2.75e-7

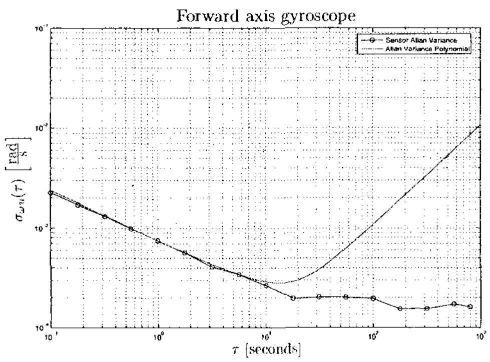
Table E.4: Forward Gyroscope Noise Coefficients

The white noise standard deviation is given by:

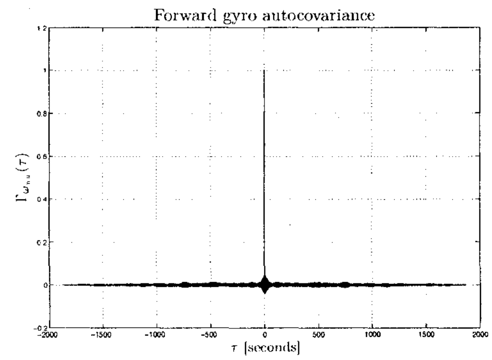
$$\sigma_{\omega u} = \frac{N_{\omega u}}{\sqrt{T}} = \frac{7.34e-4}{\sqrt{0.01}} = 0.0073\left[\frac{rad}{s}\right] \tag{E.36}$$

To determine the random walk standard deviation (which is needed for matrix "Q"), the following equation is used:

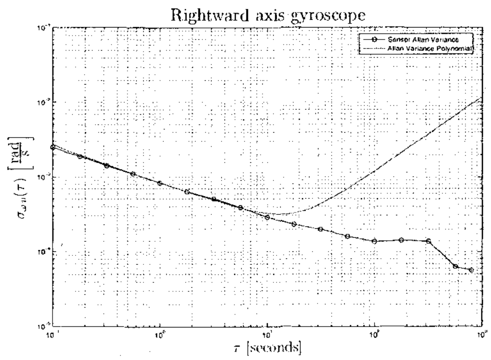
$$\sigma_{b_{\omega u}} = K_{\omega u}\sqrt{\frac{3}{T}} = 1.06e-5\sqrt{\frac{3}{0.01}} = 1.84e-4\left[\frac{rad}{s^2}\right] \tag{E.37}$$



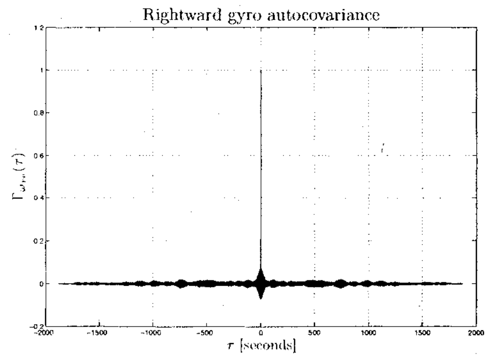
(a) Forward Allan Variance $\sigma_{\omega u}(\tau)$



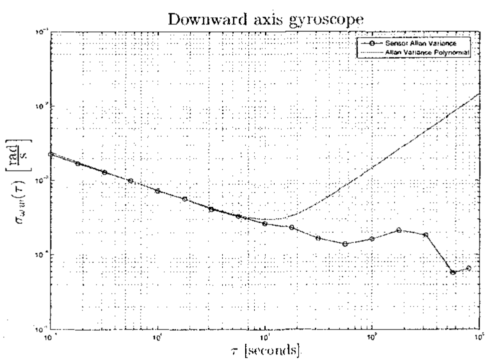
(b) Forward Auto-Covariance $\Gamma_{\omega_{uu}}(\tau)$



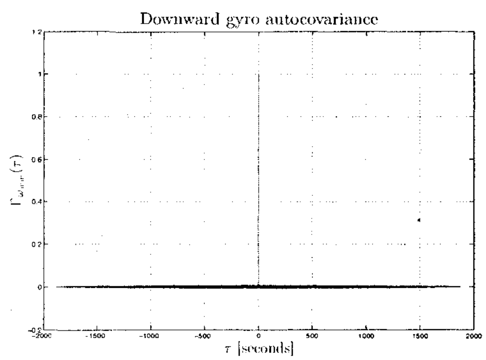
(c) Rightward Allan Variance $\sigma_{\omega v}(\tau)$



(d) Rightward Auto-Covariance $\Gamma_{\omega_{vv}}(\tau)$



(e) Downward Allan Variance $\sigma_{\omega w}(\tau)$



(f) Downward Auto-Covariance $\Gamma_{\omega_{ww}}(\tau)$

Figure E.4: Gyroscope Covariance Determination

The auto-covariance plot of the forward axis acceleration signal displayed in figure E.4(b), shows that the measurement is mainly corrupted by white noise because it looks like a sharp impulse. Therefore, it will be assumed that for the output covariance matrix "R" there will be no source of noise other than white noise.

Rightward Axis Noise Model

The allan variance plot displayed in figure E.4(c) shows roughly that after one hundred seconds the flickering noise starts corrupting the rotation measurement. Before that, white noise is the dominant source of disturbance.

Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:

$Parameter$	$Q_{\omega v}\,[rad]$	$N_{\omega v}\left[\frac{rad}{\sqrt{s}}\right]$	$B_{\omega v}\left[\frac{rad}{s}\right]$	$K_{\omega v}\left[\frac{\frac{rad}{s}}{\sqrt{s}}\right]$	$R_{\omega v}\left[\frac{rad}{s^2}\right]$
$Value$	4.33e-5	8.22e-4	2.27e-4	1.19e-5	2.34e-7

Table E.5: Rightward Gyroscope Noise Coefficients

The white noise standard deviation is given by:

$$\sigma_{\omega v} = \frac{N_{\omega v}}{\sqrt{T}} = \frac{8.22e-4}{\sqrt{0.01}} = 0.0082\left[\frac{rad}{s}\right] \tag{E.38}$$

To determine the random walk standard deviation (which is needed for matrix "Q"), the following equation is used:

$$\sigma_{b_{\omega v}} = K_{\omega v}\sqrt{\frac{3}{T}} = 1.19e-5\sqrt{\frac{3}{0.01}} = 2.07e-4\left[\frac{rad}{s^2}\right] \tag{E.39}$$

The auto-covariance plot of the forward axis acceleration signal displayed in figure E.4(d), shows that the measurement is mainly corrupted by white noise because it looks like a sharp impulse. Therefore, it will be assumed that for the output covariance matrix "R" there will be no source of noise other than white noise.

Downward Axis Noise Model

The allan variance plot displayed in figure E.4(e) shows roughly that after fifty seconds the random walk noise starts corrupting the rotation measurement.

Before that, white noise is the dominant source of disturbance.

Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:

$Parameter$	$Q_{\omega w}\,[rad]$	$N_{\omega w}\left[\frac{rad}{\sqrt{s}}\right]$	$B_{\omega w}\left[\frac{rad}{s}\right]$	$K_{\omega w}\left[\frac{\frac{rad}{s}}{\sqrt{s}}\right]$	$R_{\omega w}\left[\frac{rad}{s^2}\right]$
$Value$	3.48e-5	7.31e-4	1.91e-4	1.47e-5	4.91e-7

Table E.6: Downward Gyroscope Noise Coefficients

The white noise standard deviation is given by:

$$\sigma_{\omega w} = \frac{N_{\omega w}}{\sqrt{T}} = \frac{7.31e-4}{\sqrt{0.01}} = 0.0073\left[\frac{rad}{s}\right] \tag{E.40}$$

To determine the random walk standard deviation (which is needed for matrix "Q"), the following equation is used:

$$\sigma_{b_{\omega w}} = K_{\omega w}\sqrt{\frac{3}{T}} = 1.47e-5\sqrt{\frac{3}{0.01}} = 2.55e-4\left[\frac{rad}{s^2}\right] \tag{E.41}$$

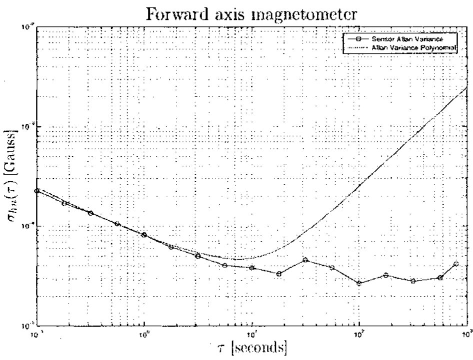
The auto-covariance plot of the forward axis acceleration signal displayed in figure E.4(f), shows that the measurement is mainly corrupted by white noise because it looks like a sharp impulse. Therefore, it will be assumed that for the output covariance matrix "R" there will be no source of noise other than white noise.

Input Covariance Submatrices

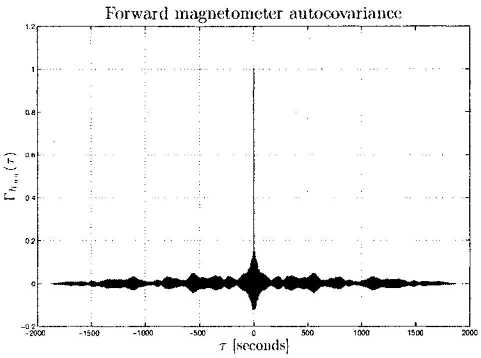
The input covariance matrices due to gyroscope measurements are:

$$Q_{\omega\omega} = \begin{bmatrix} \sigma_{\omega u}^2 & 0 & 0 \\ 0 & \sigma_{\omega v}^2 & 0 \\ 0 & 0 & \sigma_{\omega w}^2 \end{bmatrix} = \begin{bmatrix} 0.0073^2 & 0 & 0 \\ 0 & 0.0082^2 & 0 \\ 0 & 0 & 0.0073^2 \end{bmatrix} \tag{E.42}$$

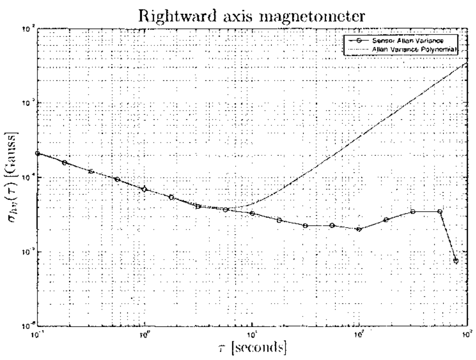
$$Q_{\omega b_\omega} = \begin{bmatrix} \sigma_{b_{\omega u}}^2 & 0 & 0 \\ 0 & \sigma_{b_{\omega v}}^2 & 0 \\ 0 & 0 & \sigma_{b_{\omega w}}^2 \end{bmatrix} = \begin{bmatrix} 3.39e-8 & 0 & 0 \\ 0 & 4.28e-8 & 0 \\ 0 & 0 & 6.5e-8 \end{bmatrix} \tag{E.43}$$



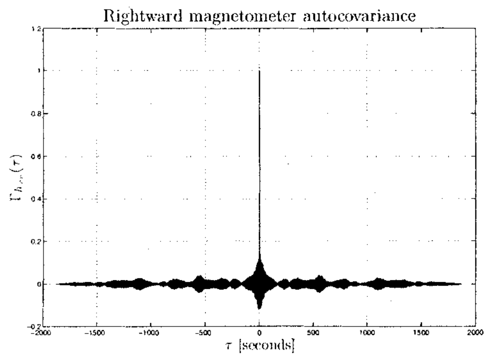
(a) Forward Allan Variance $\sigma_{\omega u}(\tau)$



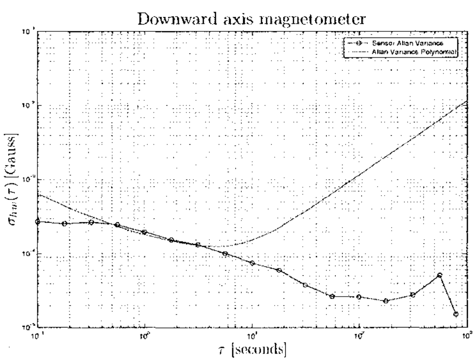
(b) Forward Auto-Covariance $\Gamma_{h_{uu}}(\tau)$



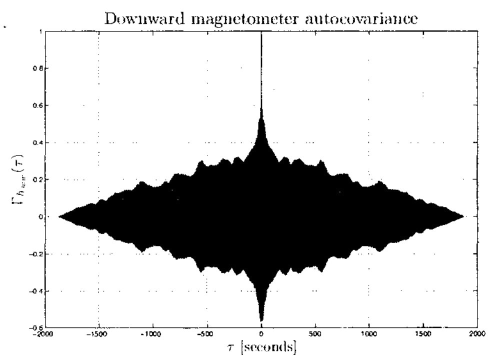
(c) Rightward Allan Variance $\sigma_{hv}(\tau)$



(d) Rightward Auto-Covariance $\Gamma_{h_{vv}}(\tau)$



(e) Downward Allan Variance $\sigma_{hw}(\tau)$



(f) Downward Auto-Covariance $\Gamma_{h_{ww}}(\tau)$

Figure E.5: Gyroscope Covariance Determination

E.3.3 Magnetometers

Figure E.5 shows the allan variance and auto-covariance plots of specific force vector data.

An important fact to point out is that the EKF designed in this project will not try to estimate the noise of the magnetometers as opposed with the gyroscopes and accelerometers. Therefore there is no need to find the input covariance matrix. It is only necessary to find the output covariance matrix.

Forward Axis Noise Model

The allan variance plot displayed in figure E.5(a) shows that during the first five seconds the measurement is mostly corrupted by white noise. After five seconds the measurement starts being disturbed by other sources of noise.

Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:

$Parameter$	$Q_{hu}\,[Gauss - s]$	$N_{hu}\,[Gauss - \sqrt{s}]$	$B_{hu}\,[Gauss]$	$K_{hu}\left[\frac{Gauss}{\sqrt{s}}\right]$	$R_{hu}\left[\frac{Gauss}{s}\right]$
$Value$	3.94e-6	7.39e-5	5.04e-5	2.5e-6	8.34e-8

Table E.7: Forward Magnetometer Noise Coefficients

The white noise standard deviation is given by:

$$\sigma_{hu} = \frac{N_{hu}}{\sqrt{T}} = \frac{7.39e-5}{\sqrt{0.01}} = 7.39e-4\,[Gauss] \tag{E.44}$$

The autocovariance plot displayed in figure E.5(b) shows that the main source of disturbance is white noise. Therefore, at the output covariance matrix there will not be any disturbance covariance other than white noise.

Rightward Axis Noise Model

The allan variance plot displayed in figure E.5(c) shows that during the first three seconds the measurement is mostly corrupted by white noise. After three seconds the measurement starts being disturbed by other sources of noise.

Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:

$Parameter$	$Q_{hv}\,[Gauss - s]$	$N_{hv}\,[Gauss - \sqrt{s}]$	$B_{hv}\,[Gauss]$	$K_{hv}\,\left[\frac{Gauss}{\sqrt{s}}\right]$	$R_{hv}\,\left[\frac{Gauss}{s}\right]$
$Value$	1.54e-6	6.8e-5	2.83e-5	3.5e-6	1.04e-7

Table E.8: Rightward Magnetometer Noise Coefficients

The white noise standard deviation is given by:

$$\sigma_{hv} = \frac{N_{hv}}{\sqrt{T}} = \frac{6.8e-5}{\sqrt{0.01}} = 6.8e-4\,[Gauss] \tag{E.45}$$

The autocovariance plot displayed in figure E.5(d) shows that the main source of disturbance is white noise. Therefore, at the output covariance matrix there will not be any disturbance covariance other than white noise.

Downward Axis Noise Model

The allan variance plot displayed in figure E.5(e) shows that during the first hundred and eighty seconds the measurement is mostly corrupted by white noise. After this the measurement starts being disturbed by Gauss-Markov process noise.

Performing a LSE over the Allan standard deviations and Solving the LSE problem the following noise constants are obtained:

$Parameter$	$Q_{hw}\,[Gauss - s]$	$N_{hw}\,[Gauss - \sqrt{s}]$	$B_{hw}\,[Gauss]$	$K_{hw}\,\left[\frac{Gauss}{\sqrt{s}}\right]$	$R_{hw}\,\left[\frac{Gauss}{s}\right]$
$Value$	2.43e-5	1.54e-4	1.36e-4	1.15e-5	3.06e-7

Table E.9: Downward Magnetometer Noise Coefficients

The white noise standard deviation is given by:

$$\sigma_{hw} = \frac{N_{hw}}{\sqrt{T}} = \frac{1.54e-4}{\sqrt{0.01}} = 0.0015\,[Gauss] \tag{E.46}$$

The autocovariance of the acceleration measurement is normalized so that $\Gamma_{h_{ww}}(\tau = 0) = 1$. A sharp impulse is seen in figure E.5(f) at $\tau = 0$. However, the curve starts decaying from $\Gamma_{h_{ww}}(\tau = 0) = 0.65$. Therefore, in order to find the autocorrelation time constant α, it suffices to find the time at which the magnitude of the autocovariance of the signal reaches $\Gamma_{h_{ww}}(\tau) = \frac{0.65}{e}$ which is $\alpha = 620s$.

To determine the Gauss-Markov standard deviation (needed for matrix "R"), the following equation is used:

$$\sigma_{GM_{hw}} = K_{hw}\sqrt{\frac{\alpha}{2}} = 1.15e-5\sqrt{\frac{620}{2}} = 2.02e-4\,[Gauss] \tag{E.47}$$

Output Covariance Submatrix

The output covariance matrix due to magnetometer measurements is:

$$R_h = \begin{bmatrix} \sigma_{hu}^2 & 0 & 0 \\ 0 & \sigma_{hv}^2 & 0 \\ 0 & 0 & \sigma_{hw}^2 + \sigma_{GM}^2 \end{bmatrix} = \begin{bmatrix} (7.39e-4)^2 & 0 & 0 \\ 0 & (6.8e-4)^2 & 0 \\ 0 & 0 & (15e-4)^2 + (2.02e-4)^2 \end{bmatrix} \tag{E.48}$$

E.3.4 Input and Output Covariance Matrices

The input covariance matrix is:

$$Q = \begin{bmatrix} Q_{a\omega} & 0 & 0 & 0 \\ 0 & Q_{\omega\omega} & 0 & 0 \\ 0 & 0 & Q_{ab_\omega} & 0 \\ 0 & 0 & 0 & Q_{\omega b_\omega} \end{bmatrix} \tag{E.49}$$

The output covariance matrix is:

$$R = \begin{bmatrix} R_{GPS} & 0 & 0 \\ 0 & R_a & 0 \\ 0 & 0 & R_h \end{bmatrix} \tag{E.50}$$

Where the GPS covariance matrix is computed by the NovAtel device online while the UAV is operating.