HØGSKOLEN
I NARVIK

SHO6267

Master of Science in Technology

Aerodynamic Modeling and Estimation of a Fixed-wing UAV

Tom Stian Andersen

July, 2013

Department of technology
Narvik University College

HØGSKOLEN
I NARVIK

Title: *Aerodynamic Modeling and Estimation of a Fixed-wing UAV*	*Date:* 22.07.2013 *Classification:* Open
Author: *Tom Stian Andersen*	*Pages:* 75 *Attachments:* 1
Department: Department of technology	
Field of Study: Satellite Engineering	
Supervisor: Ass. Prof. Raymond Kristiansen	
Principal: Narvik University College	
Principal contact: PhD. Student Espen Oland	
Keywords: UAV, Kalman, Aerodynamic, Rigid-Body, Inertial Navigation, Attitude Estimation	

Abstract (English):

An experimental platform for obtaining flight data has been developed and used to collect real flight data of a UAV. Post processing algorithms have been used to improve the results. Methods for state estimation and parameter estimation has been presented, but have not been successfully implemented. Equations of motion for an aircraft both nonlinear and linear has been derived and the dampening matrix in the linear case has been highlighted.

Abstract (Norwegian):

En eksperimentell platform for å fange flydata har blitt utviklet og brukt til å samle virkelig flydata av en UAV. Etterbehandlingsalgoritmer har blitt brukt for å forbedre resultatet. Metoder for state estimering og parameter estimering har blitt presentert, men har ikke blitt implementert med suksess. Bevegelsesligningene for et fly både ulineære og lineære har blitt utledet og dempningsmatrisen i det lineære tilfellet har blitt market ut.

Department of technology
Narvik University College

HØGSKOLEN
I NARVIK

Department of technology
Narvik University College

Høgskolen
i Narvik

Aerodynamic Modeling and Estimation of a Fixed-wing UAV

Project Description

Soon after the first flight of an unmanned aerial vehicle (UAV) in 1804 by George Cayley, the major technological challenges for unmanned aviation were defined as automatic stabilization, remote control and autonomous navigation. Since then, and especially in the last two decades, a considerable effort has been made to improve UAV technologies aiming at safety and reliability of unmanned aviation. The result is seen today as a growing use of UAV systems to perform a variety of tasks, such as military reconnaissance, geological surveys, environmental monitoring, and time-optimal search and rescue operations.

One of the main challenges with fixed-wing UAVs is the aerodynamics which must be estimated with good precision in order to obtain good control of the rigid body. This can either be done using wind tunnel testing or estimators. Using wind tunnels a scaled-down model of a fixed-wing UAV can be used to calculate the aerodynamic forces and moments acting on it at different operating conditions. These results can then be used to create models for the aerodynamics which are required for control design.

There are in general two different ways of representing the aerodynamics. One is to let the aerodynamic force vector be a function of sideslip and angle of attack and result in a fairly good aerodynamic representation. The main drawback with this method is that in order to control the UAV, it must be controlled using the angle of attack and sideslip angles in order to make the velocity components in y and z direction go to zero. Another method is to linearize the aerodynamics with regards to the linear velocity components resulting in a viscous damping matrix which is vital for control purposes of underactuated rigid bodies. This method enables a natural damping removing the requirement of controlling on the angles, and enables the system to be controlled directly on the linear velocity components. Using a linearized damping matrix in conjunction with a nonlinear damping matrix has been done for ship and AUVs, and with the same basis this should be possible to do for UAVs.

Main task

The main task of this project is to perform wind tunnel testing of the HINUAV-1 model and parameterize the aerodynamics using Taylor expansion for different operating points (i.e. changes in the orientation relative the wind vector). Furthermore the force aerodynamics shall be parameterized using linear velocity components in such a way that the natural damping becomes apparent which is critical for stability design.

Subtasks

1. Study previous work done on wind tunnel testing and different methods of parameterizing the aerodynamics such as (Mclean, 1990), (Fossen, 2011)
2. Perform wind tunnel testing of HINUAV-1
3. Use the test results and obtain coefficients for the aerodynamic forces and moments both the linear damping matrix and the nonlinear matrix
4. Augment the solution with observers to make sure that the coefficients converge to their correct values at all different states to capture additional uncertainty which may be present in the system.
5. If time allows, design an adaptive controller that captures the nonlinearities of the aerodynamics.

Summary

An experimental platform for obtaining flight data has been developed and used to collect real flight data of a UAV. Post processing algorithms have been used to improve the results. Methods for state estimation and parameter estimation has been presented, but have not been successfully implemented. Equations of motion for an aircraft both nonlinear and linear have been developed and the dampening matrix in the linear case has been highlighted.

Preface

This thesis is a requirement for obtaining a Master of Science degree in Satellite engineering at Narvik University College. The thesis presents the main results of the studies between January 2013 to July 2013 at the department of technology.

I would like to that my main supervisor Raymond Kristiansen for his excellent support and feedback, I would also like to thank my advisor Espen Oland for his invaluable guidance and support. I would also like to thank Rune Schlanbusch who also provided me with excellent support. I am very thankful of Kjell Ellingsen, Ketil Hansen, Bjarte Hoff and Jørn Bergvill for their enthusiastic and efficient assistance whenever I was in need of it. I would also like to thank Lazar Sibul for taking his time to help me constructing the experimental rig for the inertia experiment. I have also been fortunate in receiving assistance from Benyamin Akdemir who has been helping me with any issues I had while programming for this thesis, as well as offering support when implementing his previous work. I would also like to thank my fellow students Peter Smirnov, Thomas Iversen Bredeli, Andreas Jomisko and Anton Yugai, for interesting and mind opening discussions and assistance through the six months of this project. I would also like to thank the library staff and NUC, for their full support and assistance in finding the literature I was in need of, no matter how challenging it could be. I want to extend my deepest appreciation and thanks to Narvik Modellflyklubb for flying the HINUAV-1 for me and providing me with invaluable information and experience about the practical aspect of flying model airplanes. At last I would like to thank the most the one that has had to endure the most during these six months, my girlfriend Carmen, who has always supported me and motivated me to work hard.

Contents

List of Figures

1 Introduction

1.1 A brief History of aerodynamics and UAV's

There has been numerous contributors to the field of aerodynamics and maybe one of the most prolific is George Clayton. He is said to be the first aerodynamicist, born in 1773 he studied the aerodynamics of wings and used his knowledge to build the first gliders to carry a pilot. He also was the first to look into and identify the four forces that govern flight (weight, lift, drag and thrust). Derived from his work there are a lot of good literature that explain the dynamics of flight [1] and [2]. During the late 18th and whole 19th century there were numerous attempts to successfully build the first powered flying machine, but none succeeded and it remained a dream.

On December 17th, 1903 the course of history drastically changed by the flight of the first powered aerial vehicle. It was the famous Wright brothers who bravely brought the millennia long dream of human flight to our doorsteps. Since that day there have been numerous contributions to the art of flight. Since the beginning, flight has always been dependant on one crucial factor, a pilot, but recent developments have sought to phase out the human interaction with the aircraft. This new breed of airplanes are commonly referred to as Unmanned Aerial Vehicles(UAVs). UAVs have several advantages especially within scientific research, some of them being

- Smaller and lighter aircrafts,
- Cheaper and faster to build or replace,
- Missions no longer dependent on human endurance.

It was not long after the first powered human flight that ideas of unmanned aircrafts arose, but although several experiments and research projects were conducted between the first and second world war there were no major developments. It was not until the 1960s which is still remarkably early that UAVs were used successfully. The first ones called the Q2-C Firebee also known as the lightning bug [3] developed by the Ryan Aeronautical company for the US army, to be used in reconnaissance mission during the Vietnam war. The main weakness however with those UAVs was their primitive navigation algorithm and it was not until the availability of Satellite navigation systems that UAVs truly fulfilled their potential.

1.2 System identification and estimation

Parameter estimation has its origin from Karl Friedrich Gauss in 1795, which he derived from his work on estimating the position of the comet

Ceres. He postulated that the most probable values of unknown parameters are the ones that minimize the sum of the squares of the difference between observations and computed results. This is also known as the Least-Squares estimation method. Other methods has been developed over the years and can be found summarized in various literature [4] and [5]. One of the more major breakthroughs in estimation theory was in 1960, when a man known as Kalman published a paper on what is today called the Kalman filter [6]. One of the first implementations of the Kalman filter was done at the Dynamics Analysis branch at Ames Research Center at NASA. The main goal was to estimate the state vector for a manned lunar vehicle [7]. In fact since the problem was inherently nonlinear, the development of the Extended Kalman filter resulted from this work [8]. The Kalman filter approach has been studied and implemented extensively in aerospace applications [9], [10], [11] and [12]. Apart from these mentioned methods there is also Sampling Methods, which unlike the previous mentioned methods, take into account nonlinear effects on mean and covariance [13]. Although Sampling Methods are usually more computationally heavy than the former methods, recent advances has yielded a sampling method with the same computational load as the Extended Kalman Filter, called the Unscented Kalman Filter [14] and [15].

Recent advances in Micro-Electronic-Mechanical Systems (MEMS) technology has provided engineers with low cost and low weight sensors to use in strap-down inertial navigation systems. These properties give them a wide application area, in which one of them is fixed-wing UAV's. There has been a lot work in using measurements from such sensors for attitude and state estimation of small vehicles [16], [17]. In this thesis low cost sensors have been used to capture data of a fixed wing UAV in flight. Non-linear observers have then been used to extract data from the measurements which is used to estimate aerodynamic coefficients in an aerodynamic model.

1.3 Contributions and scope of the report

This thesis is an indirect part of the research project Artic EO, where it serves as a basis for the PhD program Nonlinear Control of Multiple Fixed Wing Unmanned Aerial Vehicles Flying in Formation by Espen Oland at HiN. This thesis is a continuation of a previous work done by the author at Narvik University College. The project was titled 'UAV Parameter Estimation using Recursive Least-Squares', for more insight into the project consult the digital appendix.

A similar approach to attitude estimation as shown in [18] will be used, but with real flight data and a different approach to the modelling and estimation of the aerodynamics. It was decided that instead of doing wind tunnel testing of the HINUAV-1 the work would rather focus on getting the HINUAV-1 flight ready and gather flight data for post analysis.

1.3.1 Delimitations

To decrease the workload on this thesis, the following delimitations have been introduced.

- NED coordinate frame is assumed to be sufficiently inertial for this report. The UAVs flight speed is low and it flies for a limited amount of time, meaning this assumption will hold reasonably well without deteriorating the validity of the results in this report.
- Since the UAV will be flying at low altitudes and over a relatively small area, the effects of the earths plumb-bob gravity vector will not be considered. In other words the Earth will be assumed flat.
- The UAV is assumed to be a rigid-body, meaning the distance between any two points on the UAV will remain constant.

1.4 Report outline

- Section 2: Notation of the thesis is introduced and the underlying theory that the rest of the thesis is based on is presented.
- Section 3: The Non-linear equations of motion for a fixed-wing UAV is presented as well as an linearized model with emphasis on stability derivatives. The equations are then presented in a form as to show the viscous dampening of the system in the linear case. The aerodynamic model is presented and the method to estimate its coefficients is presented. The different sensors models are presented. Two Non-linear observers are shown that will extract data from the flight test measurements.
- Section 4: The main estimation algorithms are presented.
- Section 5: The UAV and its properties are presented along with the hardware and methods used to collect the flight test data and inertia test data.
- Section 6: The experimental results are presented and a small discussion of the results and their implications.
- Section 7: The results that are obtained by employing the non-linear observers on the experimental data are shown.
- Section 8: The results from the aerodynamic estimation is presented.
- Section 9: Concluding remarks as well as suggestions for future work.

2 Theoretical background

2.1 Notation

In this report boldface will be used to denote vectors such as $\boldsymbol{v}$, scalars are represented by normal face characters and $\|\cdot\|$ represents the vector norm, this gives $\|\boldsymbol{v}\| = v$. Superscripts on vectors represents the respective frame of reference for instance $\boldsymbol{v}^b$ is the velocity vector expressed in body frame coordinates. For angular movement the subscript represents the relative angular movement for instance the vector $\boldsymbol{\omega}^b_{a,c}$ is the angular velocity of frame c relative to a expressed in b. The time derivative of a vector in an inertial frame is written $\dot{\boldsymbol{v}}^i = \frac{d\boldsymbol{v}^i}{dt}$ implying $\ddot{\boldsymbol{v}}^i = \frac{d^2\boldsymbol{v}^i}{dt^2}$, for a non inertial frame the rotation of the frame has to be accounted for $\dot{\boldsymbol{v}}^b = \frac{d\boldsymbol{v}^b}{dt} + S(\boldsymbol{\omega}^b_{i,b})\boldsymbol{v}^b$.

Rotation matrices are indicated by the letter $\boldsymbol{R}$ and $\boldsymbol{R}^b_n$ expressed the rotation from n frame to b frame. In matlab the general convention is that superscripts are always first after the variable name, then following from left right is the subscript, as an example $\boldsymbol{\omega}^b_{a,c}$ would be written w_b_ac in matlab. This convention is used consistently for all variable names. For two vectors $\boldsymbol{v}_1 = [v_1 v_2 v_3]^T, \boldsymbol{v}_2 \in \mathbb{R}$, $S(\boldsymbol{v}_1)$ is the cross-product operator and can be written as

$$S(\boldsymbol{v}_1) = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \tag{1}$$

such that $S(\boldsymbol{v}_1)\boldsymbol{v}_2 = \boldsymbol{v}_1 \times \boldsymbol{v}_2$. The vex$(\cdot)$ operator functions as the inverse of the cross-product operator, namely

$$\text{vex}(S(\boldsymbol{v}_1)) = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \tag{2}$$

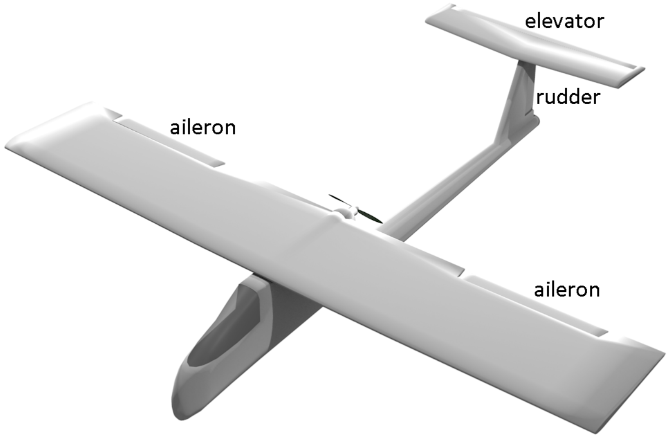

Figure 1: Illustration over the different control surfaces on the HINUAV-1

2.2 Aircraft definition

All aircrafts have control surfaces and as the name indicates, they are used to control the movement of the aircraft. In Figure 1 three types of control surfaces are shown. Ailerons are used to make the aircraft perform lateral maneuvers, for instance a barrel roll. Often when an aircraft is doing lateral maneuvers it also yaws the aircraft by using the rudder. The third control surface is the elevator which pitches the aircraft. There is also a fourth method to controlling the movement of the aircraft and that is the thrust, which accelerates the aircraft along its local x-axis. Some aircraft have additional control surfaces called flaps and are often used to brake or to make the aircraft roll faster, flaps are not implemented on the HINUAV-1 and will not be included in the thesis. In Table 1 the symbols for each of the control surfaces are shown, which are used extensively through literature.

When modelling an aircraft there are some quantities that are needed to be specified for each individual aircraft. In Figure 2, the three geometric quantities are shown

- $\bar{c}$ - Mean aerodynamic chord is the average length from the leading edge to the trailing edge of the main wing.

Table 1: Conventional control signals for an aircraft.

Thrust	δ_T
Elevator	δ_e
Aileron	δ_a
Rudder	δ_r

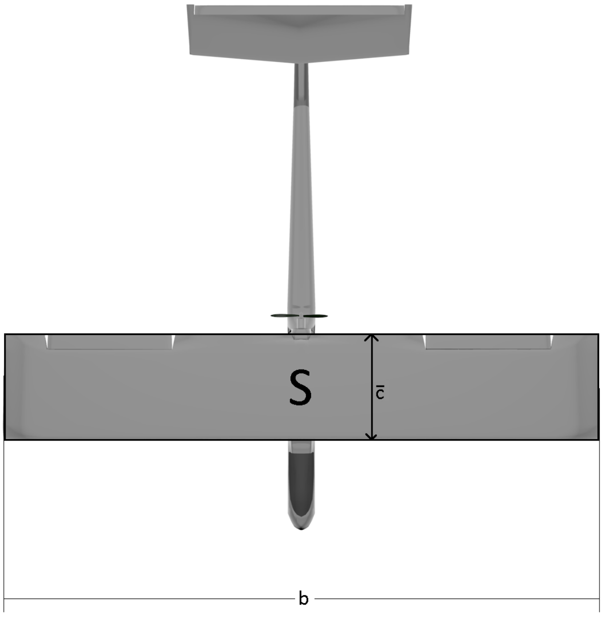

Figure 2: Illustration over the different parameters describing aircraft geometry.

- b - Wing span of entire main wing.
- S - Is the surface area of the main wing.
- m - Total mass of the aircraft.
- $\boldsymbol{J}$ - Inertia tensor, which shows how the mass is distributed on the aircraft along its principal axis.

all these need to be defined *a priori* for aerodynamic parameter estimation.

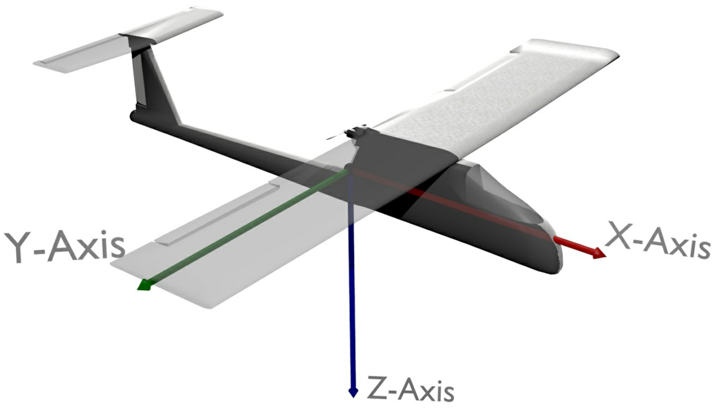

Figure 3: Body frame reference system of the HINUAV-1.

2.3 Mathematical and theoretical preliminaries

2.3.1 Coordinate frames

North East Down has the N axis pointing towards the geodetic north, the Down axis pointing down towards the center of the earth, while the East axis completes the right handed system(geodetic east).

Body frame has its center at the vehicles center of gravity, the x-axis points forward lying in the symmetric plane of the vehicle, the y-axis points through the right wing and is perpendicular to the x-axis. The z-axis completes the right handed system pointing downwards as seen in Figure 3.

Stability frame is the vehicles body frame rotated around the y-axis by the angle of attack.

Wind frame is the vehicles stability frame rotated around the z-axis by the sideslip angle. Its direction coincides and is anti-parallel with the direction of the wind.

ECEF has its center at the center of the earth, z-axis points along the earth spin axis, x-axis points through the intersection between the mean greenwich line and the equatorial plane and the y-axis completes the right handed system.

Geographic frame is essentially the same frame as ECEF but instead of cartesian coordinates it is spherical coordinates (μ, l, h) with a few distinction. The latitude(μ) in the angle between the equatorial plane and the position vector, not the angle between the spin axis and the position vector

Table 2: Symbols used for different coordinate frames.

Coordinate system	Axes
NED	N, E, D
Body	x_b, y_b, z_b
Stability	x_s, y_s, z_s
wind	x_w, y_w, z_w
ECEF	x_e, y_e, z_e
Geographic	μ, l, h

as conventional with spherical coordinates. The altitude(h) is the distance from the WGS84 reference ellipsoid to the body as show in Figure. The longitude(l) is equivalent to spherical coordinate convention. A summary of the coordinate systems is shown in Table 1.

2.3.2 Rotation between frames of reference

A rotation matrix is defined by the properties

$$R = [\boldsymbol{R} \in \mathbb{R}^{3x3};\ \boldsymbol{R}^{-1} = \boldsymbol{R}^T;\ \det \boldsymbol{R} = 1] \tag{3}$$

and it provides a means to map a vector from one frame to another. There are three widely used methods to produce such matrices.

Euler Rotations in this method one rotates a reference frame around some intermediate axis by certain angles to end up in the target reference frame. The angles and the rotation is then collected into a single matrix [19]

$$\boldsymbol{R}_B^A = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\phi s\theta c\psi - c\phi s\psi & s\phi s\theta s\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{bmatrix} \tag{4}$$

where $s(\cdot)$ and $c(\cdot)$ represents the sine and cosine operators respectively. The angles ϕ, θ and ψ are often called the roll, pitch and yaw angles respectively. Such that

$$\boldsymbol{b}^A = \boldsymbol{R}_B^A \boldsymbol{b}^B \tag{5}$$

thereby describing the relation between the two reference frames. If one of the frames are moving the rotation matrix will also change. The change in euler angles can be related to the change in the orientation of the moving frame by [20]

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{6}$$

where ω_x, ω_y and ω_z are the angular velocities of the moving frame. Euler rotations however contain singularities when solving for the euler angles, making it unsuitable to represent all type of maneuvers.

Direct Cosine Matrices are constructed by the fact that one can represent a unit vector by its angles to the coordinate axes

$$\boldsymbol{u}_a = \frac{\boldsymbol{A}}{\|\boldsymbol{A}\|} = \frac{A_x}{\|\boldsymbol{A}\|}\boldsymbol{i} + \frac{A_y}{\|\boldsymbol{A}\|}\boldsymbol{j} + \frac{A_z}{\|\boldsymbol{A}\|}\boldsymbol{k} = \cos\phi_x\boldsymbol{i} + \cos\phi_y\boldsymbol{j} + \cos\phi_z\boldsymbol{k} \quad (7)$$

using this between two reference frames will produce a direct cosine matrix

$$\boldsymbol{R}_B^A = \begin{bmatrix} \cos\phi_x & \cos\phi_y & \cos\phi_z \\ \cos\theta_x & \cos\theta_y & \cos\theta_z \\ \cos\beta_x & \cos\beta_y & \cos\beta_z \end{bmatrix} \quad (8)$$

often presented in texts as

$$\boldsymbol{R}_B^A = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \quad (9)$$

and its time derivative is defined as [21]

$$\dot{\boldsymbol{R}}_B^A = \boldsymbol{R}_B^A S(\boldsymbol{\omega}_{A,B}^B) \quad (10)$$

where $\boldsymbol{\omega}_{A,B}^B$ is the angular velocity of frame B relative to frame A, expressed in frame B. The direct cosine method contains no singularities, meaning it is suitable for all maneuvers, but at the cost of increased complexity. The Euler rotation method has three constants of integration(θ, ϕ and ψ), while the direct cosine representation has nine ($c_{ij}, \quad i,j = 1,2,...,9$).

Quaternions is based on Euler's rotation theorem that states that any rotation or composite rotation of a rigid body can be expressed as one rotation about one axis. This axis is usually termed the euler axis. If one calculates the eigenvectors of a direct cosine matrix

$$(\boldsymbol{R}_B^A - \lambda\boldsymbol{I})\boldsymbol{e} = 0 \quad (11)$$

there will be one eigenvector satisfying

$$\boldsymbol{R}_B^A\boldsymbol{e} = \boldsymbol{e} \quad (12)$$

$$\lambda = 1 \quad (13)$$

and

$$e_1^2 + e_2^2 + e_3^2 = 1 \quad (14)$$

this eigenvector lies along the euler axis. It was shown in how one can use this euler axis to construct a quaternion $\boldsymbol{q}$ with four parameters to represent any rotation in space.

$$\boldsymbol{q} = [\eta \;\; \boldsymbol{\varepsilon}] = [\eta \;\; {}_1\varepsilon \;\; {}_2\varepsilon \;\; {}_3\varepsilon]^T \tag{15}$$

$$\eta = \cos\frac{\theta}{2} \tag{16}$$

$$ {}_1\varepsilon = e_1 \cdot \sin\frac{\theta}{2} \tag{17}$$

$$ {}_2\varepsilon = e_2 \cdot \sin\frac{\theta}{2} \tag{18}$$

$$ {}_3\varepsilon = e_3 \cdot \sin\frac{\theta}{2} \tag{19}$$

and also

$$\eta^2 +_1 \varepsilon^2 +_2 \varepsilon^2 +_3 \varepsilon^2 = 1 \tag{20}$$

two quaternions representing two subsequent rotations can also be combined into one quaternion representing the whole rotation. This is done by an operation called the quaternion product

$$\boldsymbol{q}_{a,b} \otimes \boldsymbol{q}_{b,c} = \frac{1}{2} \begin{bmatrix} \eta_{a,b}\eta_{b,c} - \boldsymbol{\varepsilon}_{a,b}^T \boldsymbol{\varepsilon}_{b,c} \\ \eta_{b,c}\boldsymbol{\varepsilon}_{a,b} - \eta_{a,b}\boldsymbol{\varepsilon}_{b,c} - S(\boldsymbol{\varepsilon}_{a,b})\boldsymbol{\varepsilon}_{b,c} \end{bmatrix} = \boldsymbol{q}_{a,c} \tag{21}$$

a more intuitive representation

$$\boldsymbol{q}_{a,b} \otimes \boldsymbol{q}_{b,c} = \begin{bmatrix} \eta_{b,c} & {}_3\varepsilon_{b,c} & -{}_2\varepsilon_{b,c} & {}_1\varepsilon_{b,c} \\ -{}_3\varepsilon_{b,c} & \eta_{b,c} & {}_1\varepsilon_{b,c} & {}_2\varepsilon_{b,c} \\ {}_2\varepsilon_{b,c} & -{}_1\varepsilon_{b,c} & \eta_{b,c} & {}_3\varepsilon_{b,c} \\ -{}_1\varepsilon_{b,c} & -{}_2\varepsilon_{b,c} & -{}_3\varepsilon_{b,c} & \eta_{b,c} \end{bmatrix} \begin{bmatrix} {}_1\varepsilon_{a,b} \\ {}_2\varepsilon_{a,b} \\ {}_3\varepsilon_{a,b} \\ \eta_{a,b} \end{bmatrix} = \boldsymbol{q}_{a,c} \tag{22}$$

this can also be used for the attitude error where $\boldsymbol{q}_{a,b}$ would be the actual attitude and $\boldsymbol{q}_{b,c}$ would be there desired attitude. Quaternions can also be used to keep track of the orientation of a body through its derivative

$$\dot{\boldsymbol{q}}_{a,b} = \begin{bmatrix} \dot{\eta}_{a,b} \\ \dot{\boldsymbol{\varepsilon}}_{a,b} \end{bmatrix} = \begin{bmatrix} -\eta_{a,b}^T \cdot \boldsymbol{\omega}_{a,b}^b \\ (\eta_{a,b} \cdot I + S(\varepsilon_{a,b})) \cdot \boldsymbol{\omega}_{a,b}^b \end{bmatrix} \tag{23}$$

where $\boldsymbol{\omega}_{o,b}^b$ is the angular velocities of the body. By integrating this result you end up with a scheme to time update an orientation. The quaternions contain no singularities and only have four constants of integration making the both numerically robust and fast.

A problem with quaternions however is that the group of quaternions fulfilling (20) forms what is called a double cover on SO(3). Meaning that there are actually two eigenvectors that fulfill the requirements in (12)-(13) and (14). The eigenvector $-\boldsymbol{e}$

$$\boldsymbol{R}_B^A(-\boldsymbol{e}) = -\boldsymbol{e} \tag{24}$$

$$\lambda = 1 \tag{25}$$

and

$$(-e_1)^2 + (-e_2)^2 + (-e_3)^2 = 1 \tag{26}$$

this means that $\boldsymbol{q}$ and $-\boldsymbol{q}$ represents the same rotation, but differs where $\boldsymbol{q}$ is a rotation of θ around the axis $\boldsymbol{e}$ and $-\boldsymbol{q}$ is a rotation $-\theta$ around the axis $-\boldsymbol{e}$.

It is possible to construct a rotation matrix by using a quaternion [19]

$$\boldsymbol{R}_B^A = \boldsymbol{I}(\eta^2 - \boldsymbol{\varepsilon}^2) + 2\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T - 2\eta S(\boldsymbol{\varepsilon}) \tag{27}$$

which then can be used to rotate a vector. It is also possible to go the reverse direction

$$\eta = \pm\frac{1}{2}\sqrt{1 + c_{11} + c_{22} + c_{33}} \tag{28}$$

$${}_1\varepsilon = \frac{c_{23} - c_{32}}{4\eta} \tag{29}$$

$${}_2\varepsilon = \frac{c_{31} - c_{13}}{4\eta} \tag{30}$$

$${}_3\varepsilon = \frac{c_{12} - c_{21}}{4\eta} \tag{31}$$

which is an often used method where one keeps track of the attitude with a quaternion, but constructs a DCM, when it is needed to perform rotations.

It is important to know how to rotate between different reference frames since the dynamics of the body will be derived in inertial space, measurements will be collected in body frame and geographic frame, and the body's attitude will be the orientation between body and the inertial frame.

Geographic frame to ECEF, when given longitude l, latitude μ and altitude h the transformation is defined as [22]

$$\begin{bmatrix} x_e \\ y_e \\ z_e \end{bmatrix} = \begin{bmatrix} (N+h)\cos\mu\cos l & (N+h)\cos\mu\sin l & (\frac{a}{b}N+h)\sin\mu \end{bmatrix} \tag{32}$$

where N is the distance from the center of WGS84 ellipsoid to its surface at the respective longitude and latitude.

ECEF to NED is done by first choosing a reference vector which will be the origin of the NED frame, this is usually the take off point for aerial vehicles. Then subtracting the reference vector from all the position vectors in ECEF frame to get position coordinates relative to the NED frame. These new relative coordinates are then rotated to the NED frame

$$\boldsymbol{p}^N = \boldsymbol{R}_E^N(\boldsymbol{p}^E - \boldsymbol{p}^E_{reference}) \tag{33}$$

where $\boldsymbol{R}_E^N$ is defined as

$$\boldsymbol{R}_E^N = \begin{bmatrix} -\cos l \sin \mu & -\sin l & -\cos l \cos \mu \\ -\sin l \sin \mu & \cos l & -\sin l \cos \mu \\ \cos \mu & 0 & -\sin \mu \end{bmatrix}^T \tag{34}$$

NED to Body is the rotation matrix that describes the attitude of the body. It is usually estimated based on measurement data from various sensors onboard the body. One simple method is called the TRIAD method, where one could use two know vector quantities in the reference coordinate system and two measured vectors in the body coordinate frame, and thereby estimating the attitude by constructing two triads .

$$\begin{array}{lll} \hat{\boldsymbol{t}}_1^b = \frac{\boldsymbol{a}^b}{\|\boldsymbol{a}^b\|} & \hat{\boldsymbol{t}}_2^b = \frac{\boldsymbol{a}^b \times \boldsymbol{m}^b}{\|\boldsymbol{a}^b \times \boldsymbol{m}^b\|} & \hat{\boldsymbol{t}}_3^b = \frac{\boldsymbol{a}^b \times (\boldsymbol{a}^b \times \boldsymbol{m}^b)}{\|\boldsymbol{a}^b \times (\boldsymbol{a}^b \times \boldsymbol{m}^b)\|} \\ \hat{\boldsymbol{t}}_1^n = \frac{\boldsymbol{a}^n}{\|\boldsymbol{a}^n\|} & \hat{\boldsymbol{t}}_2^n = \frac{\boldsymbol{a}^n \times \boldsymbol{m}^n}{\|\boldsymbol{a}^n \times \boldsymbol{m}^n\|} & \hat{\boldsymbol{t}}_3^n = \frac{\boldsymbol{a}^n \times (\boldsymbol{a}^n \times \boldsymbol{m}^n)}{\|\boldsymbol{a}^n \times (\boldsymbol{a}^n \times \boldsymbol{m}^n)\|} \end{array} \tag{35}$$

where $\boldsymbol{a}^b$ is the acceleration of the body frame with the assupmtion of weak acceleration, that is the dominant acceleration of the vehicle is the gravitational acceleration, $\boldsymbol{a}^n$ is the gravity acceleration in NED frame meaning

$$\boldsymbol{a}^n = \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} \tag{36}$$

$\boldsymbol{m}^b$ is the magnetic field vector in body frame from the magnetometer onboard the body and $\boldsymbol{m}^n$ is the local magnetic field vector at the origin of the NED frame. Collecting the triad vectors in matrix form yields

$$\left[\ \hat{\boldsymbol{t}}_1^o \ \vdots \ \hat{\boldsymbol{t}}_2^o \ \vdots \ \hat{\boldsymbol{t}}_3^o\ \right] = \boldsymbol{M}_{ref} \tag{37}$$

and

$$\left[\ \hat{\boldsymbol{t}}_1^b \ \vdots \ \hat{\boldsymbol{t}}_2^b \ \vdots \ \hat{\boldsymbol{t}}_3^b\ \right] = \boldsymbol{M}_{obs} \tag{38}$$

now the estimated attitude can be written as

$$\hat{\boldsymbol{R}} = \boldsymbol{M}_{ref} \cdot \boldsymbol{M}_{obs}^T \tag{39}$$

where $\hat{\boldsymbol{R}}$ denotes that is an estimate of the true rotation matrix $\boldsymbol{R}$.

Body to Wind Transforming a vector $\boldsymbol{v}^b$ consists of first rotating the vector about the y_b axis with an angle α, this forms the stability coordinate system. Then another rotation is done about the new z_s axis by an angle β. Collecting the two rotations into one matrix gives us

$$\boldsymbol{R}_b^w = \begin{bmatrix} \cos\alpha\cos\beta & \sin\beta & \sin\alpha\cos\beta \\ -\cos\alpha\sin\beta & \cos\beta & -\sin\alpha\cos\beta \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \tag{40}$$

the new x-axis x_w now points in the opposite direction of the incoming airflow.

2.4 Taylor expansion

It is well known that if a function is continuously differentiable in an open interval which includes a point a, then the function can be expanded by a Taylor series .

$$\begin{aligned} f(x+a) &= f(a)+\frac{df(a)}{dx}(x-a)+\frac{1}{2!}\frac{d^2f(a)}{dx^2}(x-a)^2+\frac{1}{3!}\frac{d^3f(a)}{dx^3}(x-a)^3+\cdots \\ &= \sum_{n=0}^{\infty}\frac{1}{n!}\frac{d^nf(a)}{dx^n}(x-a)^n \end{aligned} \tag{41}$$

This can be used to approximate a function around a point a, by evaluating the series up to $m < \infty$

$$f(x+a) \approx \sum_{n=0}^{m}\frac{1}{n!}\frac{d^nf(a)}{dx^n}(x-a)^n \tag{42}$$

this then becomes the *mth* order Taylor polynomial. If f represents a nonlinear function, then a linear representation can be expressed by setting $m < 2$ which yields

$$f(x+a) \approx f(a) + \frac{df(a)}{dx}(x-a) \tag{43}$$

this result has a central part in linearization theory. The Maclaurin series is the Taylor series at $a = 0$ and for a function $f(x) = e^x$ can be expressed as

$$e^x = f(0) + \frac{df(0)}{dx}x + \frac{1}{2!}\frac{d^2f(0)}{dx^2}x^2 + \frac{1}{3!}\frac{d^3f(0)}{dx^3}x^3+... \tag{44}$$

It is also possible to construct the Taylor expansion of a multi-variable function as .

$$f(x_1{+}a_1,\cdots,x_n{+}a_n) = f(a_1,\cdots,a_n){+}Df(a_1,\cdots,a_n)\begin{bmatrix} x_1{-}a_1 \\ \vdots \\ x_n{-}a_n \end{bmatrix}$$

$$+\frac{1}{2!}\begin{bmatrix} x_1{-}a_1 \\ \vdots \\ x_n{-}a_n \end{bmatrix}^T D^2f(a_1,\cdots,a_n)\begin{bmatrix} x_1{-}a_1 \\ \vdots \\ x_n{-}a_n \end{bmatrix} +\ldots \qquad (45)$$

where D denotes the differential operator and is defined as

$$D = \begin{bmatrix} \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} & \cdots & \frac{\partial}{\partial x_n} \end{bmatrix} \qquad (46)$$

such that

$$Df(x_1,\cdots,x_n) = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots & \frac{\partial f}{\partial x_n} \end{bmatrix} \qquad (47)$$

now applying the differential operator a second time results in

$$D^2f(x_1,\cdots,x_n) = \begin{bmatrix} \frac{\partial^2 f}{(\partial x_1)^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{(\partial x_2)^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \\ \frac{\partial^2 f}{\partial x_n \partial x_2} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{(\partial x_n)^2} \end{bmatrix} \qquad (48)$$

which is often termed the Hessian matrix.

The second order Taylor polynomial $f(x)$ is a linear approximation of $f(x)$ at point $x = a$ and is useful when linearizing non-linear differential equations. Suppose that a set of non-linear differential equations can be written as

$$\boldsymbol{f}(x_1,\ldots,x_n) = \begin{bmatrix} f_1(x_1,\ldots,x_n) \\ f_2(x_1,\ldots,x_n) \\ \vdots \\ f_n(x_1,\ldots,x_n) \end{bmatrix} \qquad (49)$$

the linearization can then be expressed as

$$\boldsymbol{f}(x_1+a_1,\ldots,x_n+a_n) = \boldsymbol{f}(a_1,\ldots,a_n) + D\boldsymbol{f}(a_1,\cdots,a_n)\begin{bmatrix} x_1-a_1 \\ \vdots \\ x_n-a_n \end{bmatrix} \qquad (50)$$

where

$$D\boldsymbol{f}(a_1,\cdots,a_n) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \\ \frac{\partial f_n}{\partial x_n} & \frac{\partial f_n}{\partial x_n} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \qquad (51)$$

is known as the Jacobian matrix.

2.5 State space and estimation

Most dynamic systems are inherently nonlinear. A consequence of this is that tools like the laplace transform or classical control techniques cannot be used to analyse the behaviour of the system, at least not without linearization. Another problem is that a lot of systems is what is called MIMO-systems, which have multiple inputs and multiple outputs and might be of the order of 100th. If these systems were to be handled in the conventional s-plane, it would make problems more complex and therefore more difficult to solve. For this reason a new formulation called state-space was invented. In the linear case, the differential equation governing the system is reduced to first order differential equations about the systems states, the equations are then collected into a matrix format

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \tag{52}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \tag{53}$$

where $\boldsymbol{A}$ is the state matrix, $\boldsymbol{x}$ is the state vector, $\boldsymbol{B}$ is the input vector, $\boldsymbol{u}$ is the input, $\boldsymbol{y}$ is the output, $\boldsymbol{C}$ is the output matrix, $\boldsymbol{D}$ is the feed-forward matrix. This format allows the use of matrix algebra to solve the equations, which is a great advantage when it comes to digitized computing. It is important that the states in the state vector has enough entries to completely characterize the behaviour of the dynamic system. Since the dynamics are contained in (52) it is only necessary to solve (52), since (53) only maps the states and inputs to the outputs. The solution to (52) can be presented as [23]

$$\boldsymbol{x}(t) = \boldsymbol{\Phi}(t)\boldsymbol{x}(0) + \int_0^t \boldsymbol{\Phi}(t-\tau)\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \tag{54}$$

where $\boldsymbol{\Phi}(t) = e^{\boldsymbol{A}t}$, this equation can be rewritten in discrete form by specifying that the solution is sampled at constant discrete time steps($t_{k+1} - t_k = h$ and the state vector and input vector remains constant between the steps. This simplifies the convolution between the state transition matrix and the input into

$$\int_0^h e^{\boldsymbol{A}(h-\tau)} d\tau \boldsymbol{B}\boldsymbol{u}_k \tag{55}$$

which has the solution

$$\boldsymbol{A}^{-1}\left[e^{\boldsymbol{A}h} - \boldsymbol{I}\right]\boldsymbol{B} \tag{56}$$

the state space can now be written in its discrete form

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}(h)\boldsymbol{x}_k + \boldsymbol{\Delta}(h)\boldsymbol{u}_k \tag{57}$$

where $\boldsymbol{\Phi}(h) = e^{\boldsymbol{A}h}$ and $\boldsymbol{\Delta}(h) = \boldsymbol{A}^{-1}\left[\boldsymbol{\Phi}(h) - \boldsymbol{I}\right]\boldsymbol{B}$ can be calculated using the Taylor expansion of e^x

$$\boldsymbol{\Phi}(h) = \boldsymbol{I} + \boldsymbol{A}h + \frac{\boldsymbol{A}^2h^2}{2!} + \frac{\boldsymbol{A}^3h^3}{3!} + \cdots \tag{58}$$

$$\boldsymbol{A}^{-1}\left[\boldsymbol{\Phi}(h) - \boldsymbol{I}\right]\boldsymbol{B} = \left[\boldsymbol{I}h + \frac{\boldsymbol{A}h^2}{2!} + \frac{\boldsymbol{A}^2h^3}{3!} + \cdots\right]\boldsymbol{B} \tag{59}$$

and lastly the output equation is written as

$$\boldsymbol{y}_{k+1} = \boldsymbol{C}\boldsymbol{x}_{k+1} + \boldsymbol{D}\boldsymbol{u}_{k+1} \tag{60}$$

In the nonlinear case, it is not possible to separate the states and inputs from the differential equations and the state-space takes the form

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \tag{61}$$

$$\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, t) \tag{62}$$

since nonlinear systems do not conform to the superposition principle. This also means that all the tools used in linear control theory do not apply unless linearization is performed. For a quasi-linear system the linearization process can be summarized as

As mentioned earlier in this section the states should completely characterize the behaviour of the system. Therefore knowing the states, one can use this to provide the system with inputs to drive the systems to states that are desired. This realization has led to the development of observers, which will use as much available information as possible to try and estimate the values of the states that are not directly available from measurements. This task in itself can be formidable since all practical systems are subjected to noise, which can disguise the dynamics of the measured states and further complicated the estimation of the unknown ones.

2.6 Numerical differentiation and integration

In some cases there will be a need for the time derivatives of states, and if no type of sensor is employed to measure one has to rely on numerical differentiation. An example of this is angular acceleration, although instruments to measure angular acceleration exists, it is much more common to have gyroscopes that measure angular velocities instead. Another example is by GPS, which provides position data. Taking the derivative of the GPS data will give the velocities, alternatively the velocities can be obtained by numerical integration of acceleration data from an accelerometer. Once the angular velocity measurements are gathered, numerical differentiation is done to determine the angular acceleration. However a problem with take derivatives of measurements is that measurements always contain noise, and when differentiation is done without care the noise will be amplified and no useful data can be gathered from the process. One of the simplest methods of differentiation calculates the time derivative directly

$$\dot{y} \approx \frac{y(t+t_s) - y(t)}{t_s} \tag{63}$$

Often a better result can be achieved by using the difference formula

$$\dot{y} \approx \frac{y(t+t_s) - y(t-t_s)}{2t_s} \tag{64}$$

but as mentioned these methods tend to amplify the noise. A better approach is to fit a local polynomial for each data point and evaluate the derivative based on this smoothed approximation.

$$y = a_0 + a_1 t + \frac{1}{2} a_2 t^2 \tag{65}$$

which gives the time derivative

$$\dot{y} = a_1 + a_2 t \tag{66}$$

to find the coefficients a_i one can use simple least-squares estimation. If

$$z(i) = y(i) + v(i) \tag{67}$$

where $z(i)$ is the measured signal at sample time $i\delta t$ that we wish to find the time derivative of, $y(i)$ is the actual signal and $v(i)$ represents the noise. One can formulate the equations relating the coefficients to the measured data

$$\begin{aligned} a_0 + a_1(-2\delta t) + a_2(-2\delta t)^2 &= z(i-2) && (68)\\ a_0 + a_1(-\delta t) + a_2(-\delta t)^2 &= z(i-1) && (69)\\ a_0 &= z(i) && (70)\\ a_0 + a_1(\delta t) + a_2(\delta t)^2 &= z(i+1) && (71)\\ a_0 + a_1(2\delta t) + a_2(2\delta t)^2 &= z(i+2) && (72) \end{aligned}$$

Since there are 3 unknowns and 5 equation the principle of least squares applies

$$\begin{bmatrix} 1 & -2\delta t & (-2\delta t)^2 \\ 1 & -\delta t & (-\delta t)^2 \\ 1 & 0 & 0 \\ 1 & \delta t & (\delta t)^2 \\ 1 & 2\delta t & (2\delta t)^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} z(i-2) \\ z(i-1) \\ z(i) \\ z(i+1) \\ z(i+2) \end{bmatrix} \tag{73}$$

giving the least-squares solution .

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \left(\begin{bmatrix} 1 & -2\delta t & (-2\delta t)^2 \\ 1 & -\delta t & (-\delta t)^2 \\ 1 & 0 & 0 \\ 1 & \delta t & (\delta t)^2 \\ 1 & 2\delta t & (2\delta t)^2 \end{bmatrix}^T \begin{bmatrix} 1 & -2\delta t & (-2\delta t)^2 \\ 1 & -\delta t & (-\delta t)^2 \\ 1 & 0 & 0 \\ 1 & \delta t & (\delta t)^2 \\ 1 & 2\delta t & (2\delta t)^2 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & -2\delta t & (-2\delta t)^2 \\ 1 & -\delta t & (-\delta t)^2 \\ 1 & 0 & 0 \\ 1 & \delta t & (\delta t)^2 \\ 1 & 2\delta t & (2\delta t)^2 \end{bmatrix}^T \begin{bmatrix} z(i-2) \\ z(i-1) \\ z(i) \\ z(i+1) \\ z(i+2) \end{bmatrix} \tag{74}$$

As mentioned earlier velocity data can be obtained by numeric intergration of accelerometer data. A method for numerical intergration of differential equations is called Runge-Kutta where

$$\boldsymbol{y}_{i+1} = \boldsymbol{y_i} + h\Phi(t_i, \boldsymbol{y}_i, h) \tag{75}$$

where Φ is a weighted average of the derivative $\frac{d\boldsymbol{y}}{dt}$ evaluated at several points between t_i and $t_i + h$. The higher the order of the Runge-Kutta method, the more points are evaluated between t_i and $t_i + h$ and consequently, the more accurate the method becomes. In short the algorithm can be presented as [24]

$$\tilde{t}_m = t_i + a_m h \tag{76}$$

$$\tilde{\boldsymbol{y}}_m = \boldsymbol{y}_i + h \sum_{n=1}^{m-1} b_{mn} \tilde{\boldsymbol{f}}_n \tag{77}$$

$$\boldsymbol{y}_{i+1} = \boldsymbol{y_i} + h \sum_{m=1}^{s} c_m \boldsymbol{f}(\tilde{t}_m, \tilde{\boldsymbol{y}}_m) \tag{78}$$

where

$$a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_s \end{bmatrix} \quad b = \begin{bmatrix} b_{11} & & & & \\ b_{21} & b_{22} & & & \\ b_{31} & b_{32} & b_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ b_{s,1} & b_{s,2} & b_{s,3} & \dots & b_{s,s-1} \end{bmatrix} \quad c = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_s \end{bmatrix} \tag{79}$$

are the weighting coefficients.

Example RK4

$$a = \begin{bmatrix} 0 \\ \frac{1}{2} \\ \frac{1}{2} \\ 1 \end{bmatrix} \quad b = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad c = \begin{bmatrix} \frac{1}{6} \\ \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{6} \end{bmatrix} \tag{80}$$

$$\tilde{\boldsymbol{f}}_1 = \boldsymbol{f}(t_i, \boldsymbol{y}_i) \tag{81}$$

$$\tilde{\boldsymbol{f}}_2 = \boldsymbol{f}(t_i + \frac{1}{2}h, \boldsymbol{y}_i + \frac{1}{2}h\tilde{\boldsymbol{f}}_1) \tag{82}$$

$$\tilde{\boldsymbol{f}}_3 = \boldsymbol{f}(t_i + \frac{1}{2}h, \boldsymbol{y}_i + \frac{1}{2}h\tilde{\boldsymbol{f}}_2) \tag{83}$$

$$\tilde{\boldsymbol{f}}_4 = \boldsymbol{f}(t_i + h, \boldsymbol{y}_i + h\tilde{\boldsymbol{f}}_3) \tag{84}$$

$$\boldsymbol{y}_{i+1} = \boldsymbol{y}_i + h\left(\frac{1}{6}\tilde{\boldsymbol{f}}_1 + \frac{1}{3}\tilde{\boldsymbol{f}}_2 + \frac{1}{3}\tilde{\boldsymbol{f}}_3 + \frac{1}{6}\tilde{\boldsymbol{f}}_4\right) \tag{85}$$

3 Modeling

3.1 Inertia

For a detailed discussion about the derivation of the inertia matrix and its properties consult [24]. In this thesis only the main results will be considered. When working with the dynamics of rigid bodies it is important to have knowledge of the mass distribution within the body, this can be represented as a tensor

$$\boldsymbol{J} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix} \tag{86}$$

which is commonly called the inertia tensor or inertia matrix. For airplanes which are symmetric along the xz axis the inertia tensor can be simplified to

$$\boldsymbol{J} = \begin{bmatrix} I_{xx} & 0 & -I_{xz} \\ 0 & I_{yy} & 0 \\ -I_{xz} & 0 & I_{zz} \end{bmatrix} \tag{87}$$

this simplification applies to most airplanes, since airplanes tend to have symmetry about the xz axis [20]. A simple way of measuring these quantities is by suspending the aircraft in a pendulum configuration along one of its body axes as shown in Figure 4. This system can be modeled as a linearized pendulum system where the inertia around the center of mass is expressed as [18]

$$I_{CM} = \hat{m}R\left(\frac{gT^2}{4\pi^2} - R\right) - I_{rig} \tag{88}$$

this is then done for the x, y and z body axis of the aircraft. The term I_{xz} is found by measuring the oscillation period around an axis ϵ which is done by measuring the inertia of the x axis, but rotating the aircraft along the z axis by an angle ξ

$$I_{xz} = \frac{I_\xi - I_x \cos^2\xi - I_z \sin^2\xi}{\sin 2\xi} \tag{89}$$

3.2 Aircraft equations of motion

The equations of motion for a rigid-body aircraft are derived from the two classical equations

$$\sum \boldsymbol{F} = \frac{d\boldsymbol{p}}{dt} \tag{90}$$

$$\sum \boldsymbol{\tau} = \frac{d\boldsymbol{h}}{dt} \tag{91}$$

where $\boldsymbol{p}$ denotes the linear momentum vector $m\boldsymbol{v}$, $\boldsymbol{h}$ is the angular momentum vector $\boldsymbol{J\omega}$, $\boldsymbol{F}$ is the forces acting on the body and $\boldsymbol{\tau}$ is the applied

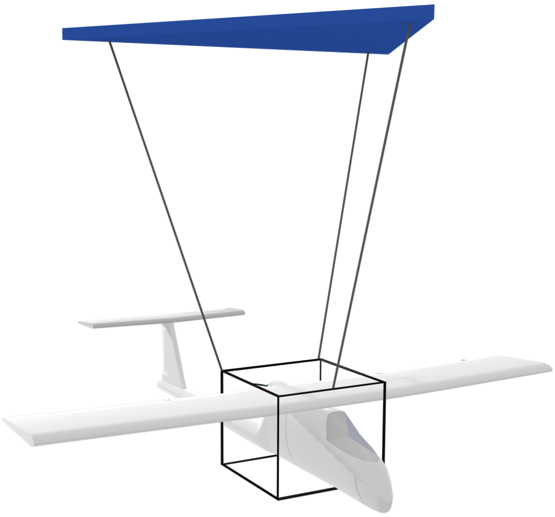

Figure 4: Illustration of the trifilar pendulum rig.

torques. The forces working on an airplane in body-frame are in general the gravitational force, the thrust and the aerodynamic forces

$$\boldsymbol{f}_g^b + \boldsymbol{f}_{thrust}^b + \boldsymbol{f}_{aero}^b = \frac{d\boldsymbol{p}^b}{dt} \tag{92}$$

$$\boldsymbol{\tau}_{aero}^b = \frac{d\boldsymbol{h}^b}{dt} \tag{93}$$

the gravitational force works on the aircrafts center of gravity and therefore does not introduce a torque, the thrust is often said to work through the x_b axis and does not produce any torque on the body. This leaves the aerodynamic forces which indeed will apply torques on the airplane by manipulation of the various control surfaces on the aircraft. Expanding the right side of (92) yields

$$\boldsymbol{f}_g^b + \boldsymbol{f}_{thrust}^b + \boldsymbol{f}_{aero}^b = \dot{\boldsymbol{p}}^b + S(\boldsymbol{\omega}_{n,b}^b)\boldsymbol{p}^b \tag{94}$$

since $\boldsymbol{p} = m\boldsymbol{v}$ and the mass of the airplane remains constant we can rewrite (94) into

$$\boldsymbol{f}_g^b + \boldsymbol{f}_{thrust}^b + \boldsymbol{f}_{aero}^b = m(\dot{\boldsymbol{v}}^b + S(\boldsymbol{\omega}_{n,b}^b)\boldsymbol{v}^b) \tag{95}$$

and solving with respect to $\dot{\boldsymbol{v}}^b$ gives

$$\dot{\boldsymbol{v}}^b = \frac{1}{m}(\boldsymbol{f}_g^b + \boldsymbol{f}_{thrust}^b + \boldsymbol{f}_{aero}^b) - S(\boldsymbol{\omega}_{n,b}^b)\boldsymbol{v}^b \tag{96}$$

Similarly with the torque equation, expanding the right side yields

$$\boldsymbol{\tau}_{aero}^b = \dot{\boldsymbol{h}}^b + S(\boldsymbol{\omega}_{n,b}^b)\boldsymbol{h}^b \tag{97}$$

since the inertia is constant (94) can be rewritten as

$$\boldsymbol{\tau}_{aero}^{b} = \boldsymbol{J}\dot{\boldsymbol{\omega}}_{n,b}^{b} + S(\boldsymbol{\omega}_{n,b}^{b})\boldsymbol{J}\boldsymbol{\omega}_{n,b}^{b} \tag{98}$$

and solving for $\dot{\boldsymbol{\omega}}_{n,b}^{b}$ yields

$$\dot{\boldsymbol{\omega}}_{n,b}^{b} = \boldsymbol{J}^{-1}(-S(\boldsymbol{\omega}_{n,b}^{b})\boldsymbol{J}\boldsymbol{\omega}_{n,b}^{b} + \boldsymbol{\tau}) \tag{99}$$

then collecting the two final equations in vector form as

$$\begin{aligned} \dot{\boldsymbol{v}}^{b} &= \frac{1}{m}(\boldsymbol{f}_{g}^{b} + \boldsymbol{f}_{thrust}^{b} + \boldsymbol{f}_{aero}^{b}) - S(\boldsymbol{\omega}_{n,b}^{b})\boldsymbol{v}^{b} && (100) \\ \dot{\boldsymbol{\omega}}_{n,b}^{b} &= \boldsymbol{J}^{-1}(-S(\boldsymbol{\omega}_{n,b}^{b})\boldsymbol{J}\boldsymbol{\omega}_{n,b}^{b} + \boldsymbol{\tau}) && (101) \end{aligned}$$

and in component form

$$\begin{aligned} \dot{u} &= \frac{1}{m}(-mg\sin\theta + T + F_x) - (qw - vr) && (102) \\ \dot{v} &= \frac{1}{m}(mg\cos\theta\sin\phi + F_y) + (pw - ur) && (103) \\ \dot{w} &= \frac{1}{m}(mg\cos\theta\cos\phi + F_z) - (pv - uq) && (104) \\ \dot{p} &= \frac{I_{xz}(\dot{r} + pq) - qr(I_{yy} - I_{zz}) + \tau_x}{I_{xx}} && (105) \\ \dot{q} &= \frac{I_{xz}(r^2 - p^2) - pr(I_{zz} - I_{xx}) + \tau_y}{I_{yy}} && (106) \\ \dot{r} &= \frac{I_{xz}(\dot{p} - qr) + pq(I_{xx} - I_{yy}) + \tau_z}{I_{zz}} && (107) \end{aligned}$$

These equations represent the nonlinear dynamics of a rigid body aircraft.

The $\boldsymbol{f}_{g}^{b}$ can be further expanded as

$$\boldsymbol{f}_{g}^{b} = m\boldsymbol{R}_{n}^{b}\boldsymbol{G}^{n} \tag{108}$$

where

$$\boldsymbol{G}^{n} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{109}$$

is the gravity acceleration ($g = 9.81$) in NED frame. The thrust can also be further expanded as

$$\boldsymbol{f}_{thrust}^{b} = \begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix} \tag{110}$$

where T for a spinning propeller can be modeled as

$$T = C_T \rho n^2 D^4 \tag{111}$$

where C_T is the thrust coefficient, ρ is the air density, n is the angular velocity of the propeller and D is the propeller diameter. It is important to keep in mind however that as the velocity of the airplane increases the thrust will decrease, because of the fact that the generated thrust depends on the difference between the incoming air to the propeller and the outcoming air of the propeller. When the velocity of the airplane increases this difference will decrease resulting in lower thrust. This can be taken into account the pressure difference in front of and behind the propeller [25]

$$p_{upstream} = p_0 + \frac{1}{2}\rho v_{air}^2 \tag{112}$$

$$p_{downstream} = p_0 + \frac{1}{2}\rho v_{exit}^2 \tag{113}$$

and by using a linear relationship between the control signal to the propeller and v_exit the propeller thrust equation can be written as

$$T = \frac{1}{2}\rho S_{prop} C_T \begin{bmatrix} (k_{motor}\delta_t)^2 - v_{air}^2 \\ 0 \\ 0 \end{bmatrix} \tag{114}$$

where S_{prop} is the area being swept by the propeller $S_{prop} \approx \frac{D^2\pi}{4}$ and k_{motor} is a constant which specifies the efficiency of the motor. The main advantage of (114) compared to (111) is the addition of the difference between incoming velocity of the air versus the outgoing velocity. The assumption of the linear relation between angular velocity and control signal also simplifies the problem of estimating the thrust since the control signal is usually known, but the rad per second of the propeller is often unknown. The coefficient C_T must either be estimated from flight data or calculated using software like for instance Propcalc which uses geometry data for the propeller to estimate C_T.

It is also possible to take the result in (100) and (101) and linearize them by the use of perturbation theory. This means that the aircraft is assumed to be in an equilibrium position and the dynamics are modeled as perturbations from the equilibrium point. The following derivation is a summary from [26]

The variables in (102)-(107) reordered to with respect to the forces

$$X = m(\dot{u} + g\sin\theta + (qw - vr)) \tag{115}$$

$$Y = m(\dot{v} - g\cos\theta\sin\phi - (pw - ur)) \tag{116}$$

$$Z = m(\dot{w} - g\cos\theta\cos\phi + (pv - uq)) \tag{117}$$

$$L = I_{xx}\dot{p} - I_{xz}(\dot{r} + pq) + qr(I_{zz} - I_{yy}) \tag{118}$$

$$M = I_{yy}\dot{q} + I_{xz}(p^2 - r^2) + pr(I_{xx} - I_{zz}) \tag{119}$$

$$N = I_{zz}\dot{r} + I_{xz}(qr - \dot{p}) + pq(I_{yy} - I_{xx}) \tag{120}$$

where F_x+T, F_y, F_z, τ_x, τ_y and τ_z has been renamed to X, Y, Z, L, M and N respectively. This is done to simplify the notation that will follow from the linearization process. The states have also been redefined to consist of a equilibrium part and a perturbation part

$$\begin{array}{ccc} u \triangleq u_0 + u & v \triangleq v_0 + v & w \triangleq w_0 + w \\ p \triangleq p_0 + p & q \triangleq q_0 + q & r \triangleq r_0 + r \\ L \triangleq L_0 + L & M \triangleq M_0 + M & N \triangleq N_0 + N \\ X \triangleq X_0 + X & Y \triangleq Y_0 + Y & Z \triangleq Z_0 + Z \end{array} \tag{121}$$

the equations representing the equilibrium position can then be represented as

$$X_0 = m(g\sin\theta_0 + (q_0w_0 - v_0r_0)) \tag{122}$$
$$Y_0 = m(-g\cos\theta_0\sin\phi_0 - (p_0w_0 - u_0r_0)) \tag{123}$$
$$Z_0 = m(-g\cos\theta_0\cos\phi_0 + (p_0v_0 - u_0q_0)) \tag{124}$$
$$L_0 = -I_{xz}p_0q_0 + q_0r_0(I_{zz} - I_{yy}) \tag{125}$$
$$M_0 = I_{xz}(p_0^2 - r_0^2) + p_0r_0(I_{xx} - I_{zz}) \tag{126}$$
$$N_0 = I_{xz}q_0r_0 + p_0q_0(I_{yy} - I_{xx}) \tag{127}$$

the perturbed equations of motions can be found by inserting (121) into (115)-(120) and subtracting (122)-(127), keeping in mind that products of squares of perturbations are small and therefore negligible such as qv or p^2, similarly for cosines of perturbed angles are approximately unity and sines of perturbed angles equals the angles themselves. Keeping this in mind results in

$$X = m(\dot{u} + q_0w + w_0q - r_0v - v_0r + g\cos\theta_0\theta) \tag{128}$$
$$Y = m(\dot{v} - p_0w - w_0p + u_0r + r_0u - g(\cos\theta_0\cos\phi_0\phi - \sin\theta_0\sin\phi_0\theta) \tag{129}$$
$$Z = m(\dot{w} + p_0v + v_0p - u_0q - q_0u + g(\cos\theta_0\sin\phi_0\phi + \sin\theta_0\cos\phi_0\theta) \tag{130}$$
$$L = I_{xx}\dot{p} - I_{xz}\dot{r} - I_{xz}(p_0q + q_0p) + (I_{zz} - I_{yy})(q_0r + r_0q) \tag{131}$$
$$M = I_{yy}\dot{q} - 2I_{xz}(r_0r - p_0p) + (I_{xx} - I_{zz})(p_0r + r_0p) \tag{132}$$
$$N = I_{zz}\dot{r} - I_{xz}\dot{p} + I_{xz}(qr_0 + rq_0) + (I_{yy} - I_{xx})(p_0q + q_0p) \tag{133}$$

it is now possible continue from these equations by expanding the left side. This is done by a Taylor approximation around the trimmed flight condition where the higher order terms are neglected. Thus the following notation is introduced .

$$\begin{array}{ccc} X_x = \frac{\partial X}{\partial x} & Y_x = \frac{\partial Y}{\partial x} & Z_x = \frac{\partial Z}{\partial x} \\ L_x = \frac{\partial L}{\partial x} & L_x = \frac{\partial M}{\partial x} & N_x = \frac{\partial N}{\partial x} \end{array} \tag{134}$$

in the expansion the influence of control variables are also included, the Taylor expansion yields .

$$\begin{aligned} X = {} & X_u u + X_{\dot{u}}\dot{u} + X_v v + X_{\dot{v}}\dot{v} + X_w w + X_{\dot{w}}\dot{w} \\ & + X_p p + X_{\dot{p}}\dot{p} + X_q q + X_{\dot{q}}\dot{q} + X_r r + X_{\dot{r}}\dot{r} \\ & + X_{\delta_T}\delta_T + X_{\dot{\delta}_T}\dot{\delta}_T + X_{\delta_e}\delta_e + X_{\dot{\delta}_e}\dot{\delta}_e \\ & + X_{\delta_a}\delta_a + X_{\dot{\delta}_a}\dot{\delta}_a + X_{\delta_r}\delta_r + X_{\dot{\delta}_r}\dot{\delta}_r \end{aligned} \tag{135}$$

$$\begin{aligned} Y = {} & Y_u u + Y_{\dot{u}}\dot{u} + Y_v v + Y_{\dot{v}}\dot{v} + Y_w w + Y_{\dot{w}}\dot{w} \\ & + Y_p p + Y_{\dot{p}}\dot{p} + Y_q q + Y_{\dot{q}}\dot{q} + Y_r r + Y_{\dot{r}}\dot{r} \\ & + Y_{\delta_T}\delta_T + Y_{\dot{\delta}_T}\dot{\delta}_T + Y_{\delta_e}\delta_e + Y_{\dot{\delta}_e}\dot{\delta}_e \\ & + Y_{\delta_a}\delta_a + Y_{\dot{\delta}_a}\dot{\delta}_a + Y_{\delta_r}\delta_r + Y_{\dot{\delta}_r}\dot{\delta}_r \end{aligned} \tag{136}$$

$$\begin{aligned} Z = {} & Z_u u + Z_{\dot{u}}\dot{u} + Z_v v + Z_{\dot{v}}\dot{v} + Z_w w + Z_{\dot{w}}\dot{w} \\ & + Z_p p + Z_{\dot{p}}\dot{p} + Z_q q + Z_{\dot{q}}\dot{q} + Z_r r + Z_{\dot{r}}\dot{r} \\ & + Z_{\delta_T}\delta_T + Z_{\dot{\delta}_T}\dot{\delta}_T + Z_{\delta_e}\delta_e + Z_{\dot{\delta}_e}\dot{\delta}_e \\ & + Z_{\delta_a}\delta_a + Z_{\dot{\delta}_a}\dot{\delta}_a + Z_{\delta_r}\delta_r + Z_{\dot{\delta}_r}\dot{\delta}_r \end{aligned} \tag{137}$$

$$\begin{aligned} L = {} & L_u u + L_{\dot{u}}\dot{u} + L_v v + L_{\dot{v}}\dot{v} + L_w w + L_{\dot{w}}\dot{w} \\ & + L_p p + L_{\dot{p}}\dot{p} + L_q q + L_{\dot{q}}\dot{q} + L_r r + L_{\dot{r}}\dot{r} \\ & + L_{\delta_T}\delta_T + L_{\dot{\delta}_T}\dot{\delta}_T + L_{\delta_e}\delta_e + L_{\dot{\delta}_e}\dot{\delta}_e \\ & + L_{\delta_a}\delta_a + L_{\dot{\delta}_a}\dot{\delta}_a + L_{\delta_r}\delta_r + L_{\dot{\delta}_r}\dot{\delta}_r \end{aligned} \tag{138}$$

$$\begin{aligned} M = {} & M_u u + M_{\dot{u}}\dot{u} + M_v v + M_{\dot{v}}\dot{v} + M_w w + M_{\dot{w}}\dot{w} \\ & + M_p p + M_{\dot{p}}\dot{p} + M_q q + M_{\dot{q}}\dot{q} + M_r r + M_{\dot{r}}\dot{r} \\ & + M_{\delta_T}\delta_T + M_{\dot{\delta}_T}\dot{\delta}_T + M_{\delta_e}\delta_e + M_{\dot{\delta}_e}\dot{\delta}_e \\ & + M_{\delta_a}\delta_a + M_{\dot{\delta}_a}\dot{\delta}_a + M_{\delta_r}\delta_r + M_{\dot{\delta}_r}\dot{\delta}_r \end{aligned} \tag{139}$$

$$\begin{aligned} N = {} & N_u u + N_{\dot{u}}\dot{u} + N_v v + N_{\dot{v}}\dot{v} + N_w w + N_{\dot{w}}\dot{w} \\ & + N_p p + N_{\dot{p}}\dot{p} + N_q q + N_{\dot{q}}\dot{q} + N_r r + N_{\dot{r}}\dot{r} \\ & + N_{\delta_T}\delta_T + N_{\dot{\delta}_T}\dot{\delta}_T + N_{\delta_e}\delta_e + N_{\dot{\delta}_e}\dot{\delta}_e \\ & + N_{\delta_a}\delta_a + N_{\dot{\delta}_a}\dot{\delta}_a + N_{\delta_r}\delta_r + N_{\dot{\delta}_r}\dot{\delta}_r \end{aligned} \tag{140}$$

collected the result together in a state space, since the UAV is of a small size and the response time of the actuators are low the actuators can be approximated to be working instantaneously, therefore neglecting the actuator dynamics yields

$$\begin{aligned}\begin{bmatrix} X \\ Y \\ Z \\ L \\ M \\ N \end{bmatrix} &= \begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ L_{\dot{u}} & L_{\dot{v}} & L_{\dot{w}} & L_{\dot{p}} & L_{\dot{q}} & L_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} \\ &+ \begin{bmatrix} X_u & X_v & X_w & X_p & X_q & X_r \\ Y_u & Y_v & Y_w & Y_p & Y_q & Y_r \\ Z_u & Z_v & Z_w & Z_p & Z_q & Z_r \\ L_u & L_v & L_w & L_p & L_q & L_r \\ M_u & M_v & M_w & M_p & M_q & M_r \\ N_u & N_v & N_w & N_p & N_q & N_r \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix} \\ &+ \begin{bmatrix} X_{\delta_T} & X_{\delta_e} & X_{\delta_a} & X_{\delta_r} \\ Y_{\delta_T} & Y_{\delta_e} & Y_{\delta_a} & Y_{\delta_r} \\ Z_{\delta_T} & Z_{\delta_e} & Z_{\delta_a} & Z_{\delta_r} \\ L_{\delta_T} & L_{\delta_e} & L_{\delta_a} & L_{\delta_r} \\ M_{\delta_T} & M_{\delta_e} & M_{\delta_a} & M_{\delta_r} \\ N_{\delta_T} & N_{\delta_e} & N_{\delta_a} & N_{\delta_r} \end{bmatrix} \begin{bmatrix} \delta_T \\ \delta_e \\ \delta_a \\ \delta_r \end{bmatrix} \end{aligned} \tag{141}$$

further on adapting the notation of that of [22] and [27]

$$\boldsymbol{\tau} = -\boldsymbol{M}_F\dot{\boldsymbol{\nu}} - \boldsymbol{N}_F\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{u} \tag{142}$$

where

$$\boldsymbol{\tau} = \begin{bmatrix} X \\ Y \\ Z \\ L \\ M \\ N \end{bmatrix} \qquad \boldsymbol{\nu} = \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix} \qquad \boldsymbol{u} = \begin{bmatrix} \delta_T \\ \delta_e \\ \delta_a \\ \delta_r \end{bmatrix}$$

$$-\boldsymbol{M}_F = \begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ L_{\dot{u}} & L_{\dot{v}} & L_{\dot{w}} & L_{\dot{p}} & L_{\dot{q}} & L_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix}$$

$$-\boldsymbol{N}_F = \begin{bmatrix} X_u & X_v & X_w & X_p & X_q & X_r \\ Y_u & Y_v & Y_w & Y_p & Y_q & Y_r \\ Z_u & Z_v & Z_w & Z_p & Z_q & Z_r \\ L_u & L_v & L_w & L_p & L_q & L_r \\ M_u & M_v & M_w & M_p & M_q & M_r \\ N_u & N_v & N_w & N_p & N_q & N_r \end{bmatrix}$$

$$\boldsymbol{B} = \begin{bmatrix} X_{\delta_T} & X_{\delta_e} & X_{\delta_a} & X_{\delta_r} \\ Y_{\delta_T} & Y_{\delta_e} & Y_{\delta_a} & Y_{\delta_r} \\ Z_{\delta_T} & Z_{\delta_e} & Z_{\delta_a} & Z_{\delta_r} \\ L_{\delta_T} & L_{\delta_e} & L_{\delta_a} & L_{\delta_r} \\ M_{\delta_T} & M_{\delta_e} & M_{\delta_a} & M_{\delta_r} \\ N_{\delta_T} & N_{\delta_e} & N_{\delta_a} & N_{\delta_r} \end{bmatrix}$$

applying the same notation to (128)-(133)

$$\boldsymbol{\tau} = \boldsymbol{M}_{RB}\dot{\boldsymbol{\nu}} + \boldsymbol{N}_{RB}\boldsymbol{\nu} + \boldsymbol{G}\boldsymbol{\eta}$$

where

$$\boldsymbol{M}_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} & 0 & -I_{xz} \\ 0 & 0 & 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & -I_{xz} & 0 & I_{zz} \end{bmatrix}$$

$$\boldsymbol{G} = \begin{bmatrix} 0 & 0 & 0 & 0 & mg\cos\theta_0 & 0 \\ 0 & 0 & 0 & -mg\cos\theta_0\cos\phi_0 & mg\sin\theta_0\sin\phi_0 & 0 \\ 0 & 0 & 0 & mg\cos\theta_0\sin\phi_0 & mg\sin\theta_0\cos\phi_0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{\eta} = \begin{bmatrix} x_n \\ y_n \\ z_n \\ \phi \\ \theta \\ \psi \end{bmatrix}$$

.

$$\boldsymbol{N}_{RB} = \begin{bmatrix} 0 & -mr_0 & mq_0 & 0 & mw_0 & -mv_0 \\ mr_0 & 0 & -mp_0 & -mw_0 & 0 & mu_0 \\ -mq_0 & mp_0 & 0 & mv_0 & -mu_0 & 0 \\ 0 & 0 & 0 & -I_{xz}q_0 & -I_{xz}p_0+(I_{zz}-I_{yy})r_0 & (I_{zz}-I_{yy})q_0 \\ 0 & 0 & 0 & 2I_{xz}p_0+(I_{xx}-I_{zz})r_0 & 0 & -2I_{xz}r_0+(I_{xx}-I_{zz})p_0 \\ 0 & 0 & 0 & (I_{yy}-I_{xx})q_0 & I_{xz}r_0+(I_{xx}-I_{xx})p_0 & I_{xz}q_0 \end{bmatrix}$$

we can now collect everything into one equation

$$\boldsymbol{M}_{RB}\dot{\boldsymbol{\nu}} + \boldsymbol{N}_{RB}\boldsymbol{\nu} + \boldsymbol{G}\boldsymbol{\eta} = -\boldsymbol{M}_F\dot{\boldsymbol{\nu}} - \boldsymbol{N}_F\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{u} \tag{143}$$

rearranging yields

$$(\boldsymbol{M}_{RB} + \boldsymbol{M}_F)\dot{\boldsymbol{\nu}} + (\boldsymbol{N}_{RB} + \boldsymbol{N}_F)\boldsymbol{\nu} + \boldsymbol{G}\boldsymbol{\eta} = \boldsymbol{B}\boldsymbol{u} \tag{144}$$

and finally solving for $\dot{\boldsymbol{\nu}}$.

$$\begin{aligned} \dot{\boldsymbol{\nu}} =& (\boldsymbol{M}_{RB}+\boldsymbol{M}_F)^{-1}\boldsymbol{B}\boldsymbol{u} \\ & -(\boldsymbol{M}_{RB}+\boldsymbol{M}_F)^{-1}(\boldsymbol{N}_{RB}+\boldsymbol{N}_F)\boldsymbol{\nu} - (\boldsymbol{M}_{RB}+\boldsymbol{M}_F)^{-1}\boldsymbol{G}\boldsymbol{\eta} \end{aligned} \tag{145}$$

the matrix $\boldsymbol{N}_F$ is referred to as the linear dampening matrix. It is also possible to further simplify the equations by separating the longitudal and lateral dynamics and the fact that several of the stability derivatives can be neglected. Consult [26] for more information on this topic.

3.3 Aerodynamics

For UAV application a good model for aerodynamics is very important. Aerodynamics describe the relation between the incoming airstream and how it translates to forces and torques acting on the aircraft. There are many ways and methods to model this relation. One model was inadvertently presented during the linearization of the aircraft dynamics namely (141). Most of the dimensional derivatives in (141) can be neglected for a conventional aircraft resulting in [20]

$$X = X_u u + X_w w + X_q q + X_\delta \delta \tag{146}$$
$$Z = Z_u u + Z_w w + Z_{\dot{w}} \dot{w} + Z_q q + Z_\delta \delta \tag{147}$$
$$M = M_u u + M_w w + M_{\dot{w}} \dot{w} + M_q q + M_\delta \delta \tag{148}$$

$$Y = Y_v v + Y_p p + Y_r r + Y_\delta \delta \tag{149}$$
$$L = L_v v + L_p p + L_r r + L_\delta \delta \tag{150}$$
$$N = N_v v + N_p p + N_r r + N_\delta \delta \tag{151}$$

where the equations have been divided into longitudal (X, Z, M) and lateral (Y, L, N) forces.

For an aircraft there are in general two measurements of velocity, ground velocity $\boldsymbol{V_g}$ and velocity relative to the surrounding air $\boldsymbol{V}_{air}$. The ground velocity is a quantity often given by GPS-receivers, while air speed is measured with an instrument called a pitot tube. They are related together through

$$\boldsymbol{v}_{air}^b = \boldsymbol{R}_n^b(\boldsymbol{v}_g^n - \boldsymbol{v}_w^n) \tag{152}$$

where $\boldsymbol{v}_w^n$ is the wind vector expressed in NED frame. If its a calm day and no wind is present then

$$\boldsymbol{v}_g^b = \boldsymbol{v}_{air}^b \tag{153}$$

when this relation holds the angle of attack and sideslip can be expressed as

$$\alpha = \tan^{-1} \frac{w}{u} \tag{154}$$
$$\beta = \sin^{-1} \frac{v}{\|\boldsymbol{v}^b\|} \tag{155}$$

often the aerodynamic models derived from wind tunnel tests are dependent on the angle of attack and sideslip angles. The most common means of modeling aerodynamic relationships is through Taylor expansions equivalent to that of (135)-(140). The expansions in (135)-(140) are called dimensional derivatives, it is possible to express them as nondimensional aerodynamic coefficients [20]

$$C_X = C_{X_0} + C_{X_u}\frac{\Delta u}{u_0} + C_{X_w}\frac{\Delta w}{u_0} + C_{X_q}\frac{q\bar{c}}{2u_0} + C_{X_\delta}\Delta\delta \tag{156}$$

$$C_Y = C_{Y_0} + C_{Y_\beta}\Delta\beta + C_{Y_p}\frac{pb}{2V_0} + C_{Y_r}\frac{rb}{2V_0} + C_{Y_\delta}\Delta\delta \tag{157}$$

$$C_Z = C_{Z_0} + C_{Z_u}\frac{\Delta u}{u_0} + C_{Z_w}\frac{\Delta w}{u_0} + C_{Z_{\dot{w}}}\left[\frac{\dot{w}}{u_0}\right]\frac{\bar{c}}{2u_0} + C_{Z_q}\frac{q\bar{c}}{2u_0} + C_{Z_\delta}\Delta\delta \tag{158}$$

$$C_l = C_{l_0} + C_{l_\beta}\Delta\beta + C_{l_p}\frac{pb}{2V_0} + C_{l_r}\frac{rb}{2V_0} + C_{l_\delta}\Delta\delta \tag{159}$$

$$C_m = C_{m_0} + C_{m_u}\frac{\Delta u}{u_0} + C_{m_w}\frac{\Delta w}{u_0} + C_{\dot{w}}\left[\frac{\dot{w}}{u_0}\right]\frac{\bar{c}}{2u_0} + C_{m_q}\frac{q\bar{c}}{2u_0} + C_{m_\delta}\Delta\delta \tag{160}$$

$$C_n = C_{n_0} + C_{n_\beta}\Delta\beta + C_{n_p}\frac{pb}{2V_0} + C_{n_r}\frac{rb}{2V_0} + C_{n_\delta}\Delta\delta \tag{161}$$

where Δ signifies the difference between the equilibrium state and perturbed state i.e. $\Delta u = u_0 - u$. The model presented in (156)-(161) is linear and is only valid with small perturbations from the aircrafts equilibrium state. It is also possible to define a nonlinear model for the aerodynamics by including higher order terms from the Taylor expansion, or by letting the dimensionless derivatives become functions of the states. Modelling the aerodynamics of a aircraft is a formidable task and the model structure will depend on the type of aircraft and its intended goal. Aerodynamic models range from simplified linear equations for fix-winged UAVs [18] to global nonlinear equations for F-16 aircrafts [28].

3.4 Sensor models

3.4.1 Rate gyros

Most commonly rate gyros in a strapdown configuration measure the angular velocity of the vehicles body frame with respect to an inertial frame, expressed in the vehicles body frame. Rate gyros are usually affected by noise and a constant bias, which is why integrating rate gyro measurements will lead to divergence of the attitude over time. The error model for a typical rate gyro can then be expressed as [16]

$${}_m\boldsymbol{\omega}_{i,b}^b = {}_t\boldsymbol{\omega}_{i,b}^b + \boldsymbol{b}_\omega + \boldsymbol{v}_\omega \tag{162}$$

where $_m$ denotes the measured angular velocity, $_t$ denotes the true angular velocity, $\boldsymbol{b}_\omega$ is the gyro bias and $\boldsymbol{v}_\omega$ is the additive measurement noise.

3.4.2 Accelerometer

Accelerometers in a strapdown configuration measure the translational acceleration of the vehicle in body frame. The translational force acting on a mass can be expressed as [29]

$$F = ma = m\zeta + mg \tag{163}$$

where F is the total force, m is the mass, ζ is the acceleration produced by other forces than the gravitational field and g is the gravitational acceleration. The inside of a small accelerometer can be expressed as a small proof mass connected to a cage by two springs on either side of the mass as shown in Figure 5. When the accelerometer is accelerated the mass is displaced from its equilibrium position, this displacement is measured by the displacement of the needle on the displacement pick-off. When the accelerometer sits on the table, the sum of the forces acting on the accelerometer is zero which means according to (163)

$$-m\zeta = mg \tag{164}$$

therefore the output of the accelerometer will be equal to $\zeta = -g$. Similarly if the accelerometer is in free-fall, the mass will be at its equilibrium position and $f = 0$, which results in

$$a = g \tag{165}$$

following from this the output equation from the accelerometer can be expressed as

$$a = \zeta - g \tag{166}$$

or in vector form in a strapdown configuration

$$\boldsymbol{a}^b = \boldsymbol{\zeta}^b - \boldsymbol{g}^b \tag{167}$$

but knowing from (100) that

$$\dot{\boldsymbol{v}}^b + S(\boldsymbol{\omega}_{n,b}^b)\boldsymbol{v}^b - \frac{\boldsymbol{f}_g^b}{m} = \frac{1}{m}(\boldsymbol{f}_{thrust}^b + \boldsymbol{f}_{aero}^b) \tag{168}$$

we can rewrite the output equation for the accelerometer as [20]

$$\boldsymbol{a}^b = \frac{1}{m}(\boldsymbol{f}_{thrust}^b + \boldsymbol{f}_{aero}^b) \tag{169}$$

the error model for the accelerometer can be expressed as [16]

$${}_m\boldsymbol{a}^b = {}_t\boldsymbol{a}^b + \boldsymbol{b}_a + \boldsymbol{v}_a \tag{170}$$

where $_m$ denotes the measured acceleration, $_t$ denotes the true acceleration, $\boldsymbol{b}_a$ is the accelerometer bias and $\boldsymbol{v}_a$ is the additive measurement noise.

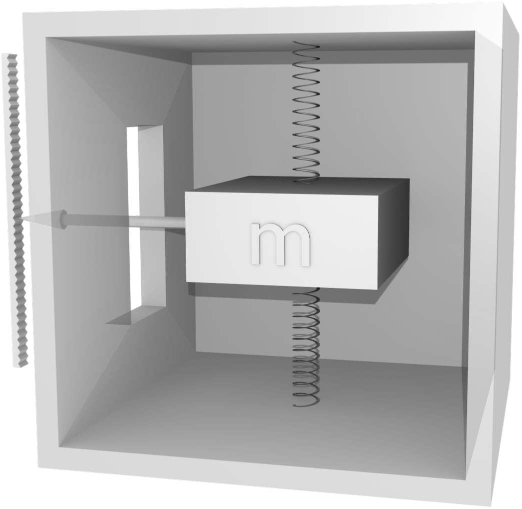

Figure 5: Illustration of a conceptual accelerometer

3.4.3 Magnetometer

A magnetometer measures the magnetic field vector in body frame, given by [30]

$$\boldsymbol{m}^b = \boldsymbol{R}_n^b \begin{bmatrix} \cos\varphi \cos\chi \\ \sin\varphi \cos\chi \\ \sin\chi \end{bmatrix} B^n \tag{171}$$

where B^n is the magnitude of the flux density, φ is the declination angle and χ is the inclination angle. The magnetic field vector can also be found in NED frame as a function of its reference position in geographic coordinates

$$\boldsymbol{m}^n(\boldsymbol{p}_{ref}^g) = \begin{bmatrix} \cos\varphi(\boldsymbol{p}_{ref}^g) \cos\chi(\boldsymbol{p}_{ref}^g) \\ \sin\varphi(\boldsymbol{p}_{ref}^g) \cos\chi(\boldsymbol{p}_{ref}^g) \\ \sin\chi(\boldsymbol{p}_{ref}^g) \end{bmatrix} B(\boldsymbol{p}_{ref}^g) \tag{172}$$

where $\boldsymbol{p}_{ref}^g = [l, \mu, h]$. For attitude estimation the magnitude of the magnetic field vector is of no importance and the vector is usually normalized

$$\bar{\boldsymbol{m}}^b = \frac{\boldsymbol{m}^b}{\|\boldsymbol{m}^b\|} \tag{173}$$

for a typical magnetometer (171) can be extended to include noise and disturbances as found in [16]

$${}_m\boldsymbol{m}^b = \boldsymbol{R}_n^b \boldsymbol{m}^n + \boldsymbol{B}_m + \boldsymbol{v}_m \tag{174}$$

where $\boldsymbol{B}_m$ is the local magnetic disturbances in body-fixed frame. Although the magnetometer gives the aircrafts orientation relative to the magnetic field, it is not enough to determine the attitude of the aircraft. This can easily be show by using a small rotation in the body axes [31]

$$\begin{bmatrix} H'_x \\ H'_y \\ H'_z \end{bmatrix} = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix} \tag{175}$$

where $[H_x \;\; H_y \;\; H_z]^T$ is the new direction of the magnetic field vector after a small rotation. For small rotation it holds that $\cos(\cdot) \approx 1$ and $\sin(\cdot) \approx (\cdot)$, this simplifies the rotation into

$$\begin{bmatrix} H'_x \\ H'_y \\ H'_z \end{bmatrix} = \begin{bmatrix} 1 & -\psi & \theta \\ \psi & 1 & -\phi \\ -\theta & \phi & 1 \end{bmatrix} \begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix} \tag{176}$$

and this can be written as

$$\begin{bmatrix} H'_x \\ H'_y \\ H'_z \end{bmatrix} = \left(\boldsymbol{I} + \begin{bmatrix} 0 & -\psi & \theta \\ \psi & 0 & -\phi \\ -\theta & \phi & 0 \end{bmatrix} \right) \begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix} \tag{177}$$

the equation can now be rewritten in terms of the incremental rotation $\Delta H'_x$

$$\begin{bmatrix} \Delta H'_x \\ \Delta H'_y \\ \Delta H'_z \end{bmatrix} = \begin{bmatrix} 0 & -\psi & \theta \\ \psi & 0 & -\phi \\ -\theta & \phi & 0 \end{bmatrix} \begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix} \tag{178}$$

since the rotation matrix is singular there is no unique solution with respect to the euler angles $[\phi \;\; \theta \;\; \psi]$. At most two of the angles can be known while the third will a free variable, another vector measurement is needed.

3.4.4 GPS

GPS measurements have a tendency to be bias free. The only drawback with GPS is its precision. If one leaves a GPS receiver in a fixed position for an extended period of time, the measured position will be stable and not drift, but the measurements will not improve over time. In this thesis GPS measurements will be modeled as

$${}_m\boldsymbol{p}^n = {}_t\boldsymbol{p}^n + \boldsymbol{v}_{gps} \tag{179}$$

and the velocity will be given as the derivative of the position

$${}_m\boldsymbol{v}^n = {}_m\dot{\boldsymbol{p}}^n \tag{180}$$

3.4.5 Pressure sensor

There are mainly two types of pressure sensors which have important applications in aerospace, those being the absolute pressure sensor and the differential pressure sensor.

The absolute pressure sensor measures the absolute pressure often expressed as kilo Pascal. The measurements can be used to estimate the altitude of the aircraft or to estimate the dynamic pressure $\bar{q}$. The sensors are usually have a temperature related bias drift and is subjected to measurement noise and can be represented by the equation [25]

$$_{m}p = \ _{t}p + B_p + v_p \tag{181}$$

where $\boldsymbol{v}_p$ is additive noise. Since the pressure reduces with attitude it can be used to calculate altitude by using the relation between pressure and altitude [25]

$$p = p_0 \left[\frac{T_0}{T_0 + L_0 h_{asl}} \right]^{\frac{gM}{RL_0}} \tag{182}$$

where p_0 is the standard pressure at sea level, h_{asl} is altitude above sea level, L_0 is the lapse rate, g is the gravitational constant, R is the universal gas constant for air, T_0 is the standard temperature at sea level and M is the standard molar mass of atmospheric air. Solving this equation with respect to altitude yields

$$h_{asl} = \frac{T_0}{L_0} \left(\frac{p_0}{p} \right)^{\frac{RL_0}{gM}} - \frac{T_0}{L_0} \tag{183}$$

It is important to keep in mind that this altitude is with respect to the sea level and will differ from the altitude from the GPS which is with respect to the WGS84 ellipsoid.

Differential pressure sensors are used together with a pitot-tube to measure the aircrafts airspeed. The pitot-tube must be placed clear of the aircraft body to prevent the body from disturbing the airflow to the pitot-tube. The pitot static-tube sensor works by measuring the difference between the dynamic and static pressure and then using bernoulli's equation the airspeed can be calculated [12]

$$V_{pitot}^2 = K \frac{2\Delta P}{\rho} \tag{184}$$

where Δp is the dynamic pressure from the pitot-tube sensor, ρ is the air density and K is a correction factor.

In [12] it was demonstrated that it is possible to estimate the wind direction and speed by using a pitot-tube and a GPS-receiver. By using the relation from (152)

$$\boldsymbol{v}_w^n \approx \boldsymbol{v}_g - \boldsymbol{R}_b^n \begin{bmatrix} 1 \\ \beta \\ \alpha \end{bmatrix} V_{pitot} \tag{185}$$

where $\boldsymbol{v}_a$ has been approximated to be

$$\boldsymbol{v}_a^b \approx \begin{bmatrix} 1 \\ \beta \\ \alpha \end{bmatrix} V_{pitot} \tag{186}$$

which holds for small angles of α and β. To show this consider the relation between speed relative to surrounding air and pitot speed [12]

$$V_{pitot}^2 = \|\boldsymbol{v}_{air}\|^2 \cos\alpha \cos\beta \tag{187}$$

$$\|\boldsymbol{v}_{air}\|^2 = \frac{V_{pitot}^2}{\cos\alpha\cos\beta} \tag{188}$$

$$\tag{189}$$

but from (186) we have that

$$\|\boldsymbol{v}_{air}^b\|^2 = V_{pitot}^2(1 + \beta^2 + \alpha^2) \tag{190}$$

this gives

$$\frac{1}{\cos\alpha\cos\beta} = (1 + \beta^2 + \alpha^2) \tag{191}$$

$\cos\alpha \approx 1$ for small angles of α, which also means that $\alpha^2 \approx 0$ and the same holds for the β angle.

It shares the same model structure as the absolute pressure sensor, with additive noise and a temperature related bias drift [25]

$$_m p = \ _t p + B_p + v_p \tag{192}$$

4 State and parameter estimation

4.1 Kalman filter

The Kalman filter is often referred to as an optimal observer. It estimates the states of a linear system in presence of noise. Like most observers and estimators it is made up of a prediction and correction term. What is interesting about the Kalman filter is that it minimizes the variance of the state error

$$\tilde{x} = x - \hat{x} \tag{193}$$

where $\tilde{x}$ is the state error, x is the state and $\hat{x}$ is the estimated state.

In most literature it is the discrete time Kalman filter which is explained since its simpler and is usually the algorithm that is implemented. The prediction step for a luenberger observer is given as [23]

$$\hat{\boldsymbol{x}}_{k+1|k} = \boldsymbol{\Phi}\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{\Delta}\boldsymbol{u}_k \tag{194}$$

$$\hat{\boldsymbol{z}}_{k+1} = \boldsymbol{C}\hat{\boldsymbol{x}}_{k+1|k} \tag{195}$$

and the correction step is given as [23]

$$\tilde{\boldsymbol{z}}_{k+1} = \boldsymbol{z}_{k+1} - \hat{\boldsymbol{z}}_{k+1} \tag{196}$$

$$\hat{\boldsymbol{x}}_{k+1|k+1} = \hat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}\tilde{\boldsymbol{z}}_{k+1} \tag{197}$$

the Kalman filter is similar in structure to the luenberger but the main difference is the gain matrix $\boldsymbol{K}$ which is used to determine how much the state estimates should be corrected. For a Kalman filter this gain matrix is determined such that the covariance matrix $\boldsymbol{P} = E[\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^T]$ is minimized. One of the main difference between the Kalman Filter and Luenberger observer aside from the gain matrix $\boldsymbol{K}$, is that the Kalman Filter assumes the system is corrupted with noise. Consider the system in (52)-(53) with no feed forward and with added noise

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{\Gamma}\boldsymbol{w}_k \tag{198}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{v}_k \tag{199}$$

Assume that $\boldsymbol{v}$ and $\boldsymbol{w}$ are white noise with $E[\boldsymbol{v}] = 0$ and $E[\boldsymbol{w}] = 0$, then the full algorithm can be expressed as the following equations [23]

$$\boldsymbol{P}_k^* = \boldsymbol{\Phi}\boldsymbol{P}_k\boldsymbol{\Phi}^T + \boldsymbol{\Gamma}\boldsymbol{Q}\boldsymbol{\Gamma}^T \tag{200}$$

$$\boldsymbol{K}_{k+1} = \boldsymbol{P}_k^*\boldsymbol{C}^T\left[\boldsymbol{C}\boldsymbol{P}_k^*\boldsymbol{C}^T + \boldsymbol{R}\right]^{-1} \tag{201}$$

$$\hat{\boldsymbol{x}}_{k+1|k+1} = \left[\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}\right]\left[\boldsymbol{\Phi}\hat{\boldsymbol{x}}_{k|k} + \boldsymbol{\Delta}\boldsymbol{u}_k\right] + \boldsymbol{K}_{k+1}\boldsymbol{z}_{k+1} \tag{202}$$

$$\boldsymbol{P}_{k+1} = \left[\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}\right]\boldsymbol{P}_k^* \tag{203}$$

where $\boldsymbol{R} = E[\boldsymbol{v}\boldsymbol{v}^T]$ and $\boldsymbol{Q} = E[\boldsymbol{w}\boldsymbol{w}^T]$. For more detail on the derivation process consult [23]

4.2 Extended Kalman filter

One of the main drawbacks of the kalman filter is that it only applies to linear systems while most dynamic systems are nonlinear. The Extended Kalman Filter is intended to bridge the gap, at least for systems that are near linear. The Extended Kalman filter linearizes the system along the estimated trajectory and evaluates the covariance and correction gain matrix from the linearized system. The estimated states and output are calculated using the full nonlinear equations. The first step is extrapolation/prediction

$$\hat{\boldsymbol{x}}_{k+1|k} = \hat{\boldsymbol{x}}_{k|k} + \int_{t_k}^{t_{k+1}} \boldsymbol{f}[\hat{\boldsymbol{x}}_{k|k}, \bar{\boldsymbol{u}}_k]dt \tag{204}$$

$$\boldsymbol{P}_k^* \approx \boldsymbol{\Phi}_k \boldsymbol{P}_k \boldsymbol{\Phi}_k^T + \boldsymbol{\Gamma}\boldsymbol{Q}\boldsymbol{\Gamma}^T \tag{205}$$

the next step is the update/correction step

$$\hat{\boldsymbol{y}}_{k+1} = \boldsymbol{h}[\hat{\boldsymbol{x}}_{k+1|k}, \boldsymbol{u}_k] \tag{206}$$

$$\boldsymbol{K}_{k+1} = \boldsymbol{P}_k^* \boldsymbol{C}^T[\boldsymbol{C}\boldsymbol{P}_k^*\boldsymbol{C}^T + \boldsymbol{R}]^{-1} \tag{207}$$

$$\hat{\boldsymbol{x}}_{k+1|k+1} = \hat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{K}_k[\boldsymbol{z}_k - \hat{\boldsymbol{y}}_{k+1}] \tag{208}$$

$$\boldsymbol{P}_{k+1} = [\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}]\boldsymbol{P}_k^* \tag{209}$$

these equations are under the assumption that the model and measurement noise are additive and is not included in the nonlinear model. It is clear that there are similarities between the Extended and the Linear Kalman Filters. This algorithm also requires the linearization of the state equations and output equations defined as

$$\boldsymbol{C} \approx \frac{d\boldsymbol{h}}{d\boldsymbol{x}} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \frac{\partial h_2}{\partial x_1} & \frac{\partial h_2}{\partial x_2} & \cdots & \frac{\partial h_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \\ \frac{\partial h_n}{\partial x_n} & \frac{\partial h_n}{\partial x_n} & \cdots & \frac{\partial h_n}{\partial x_n} \end{bmatrix} \tag{210}$$

and

$$\boldsymbol{A} \approx \frac{d\boldsymbol{f}}{d\boldsymbol{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \\ \frac{\partial f_n}{\partial x_n} & \frac{\partial f_n}{\partial x_n} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \tag{211}$$

where $\boldsymbol{\Phi}$ is found from $\boldsymbol{A}$ by using (58). The integral in (204) can be evaluated using Runge-Kutta integration.

4.3 Explicit complementary filter

The explicit complementary filter assumes the aircraft is under weak acceleration and therefore the gravitational acceleration is dominant. In other words it assumes that $\frac{\boldsymbol{a}^b}{\|\boldsymbol{a}^b\|} \approx -\boldsymbol{R}_n^b \boldsymbol{e}_3$. In addition to an accelerometer the aircraft needs to have a magnetometer and a 3-axis gyro. The accelerometer and magnetometer is used to estimate the attitude of the aircraft while the gyro's are used to improve the estimate. The observer can be presented as [16]

$$\boldsymbol{\omega}_{mes} = -vex\left(\sum_{i=1}^{n} \frac{k_i}{2}(\boldsymbol{v}_i \hat{\boldsymbol{v}}_i^T - \hat{\boldsymbol{v}}_i \boldsymbol{v}_i^T))\right) \tag{212}$$

$$\dot{\hat{\boldsymbol{q}}} = \frac{1}{2}\hat{\boldsymbol{q}} \otimes \boldsymbol{p}(\boldsymbol{\omega}_{n,b}^b - \hat{\boldsymbol{b}} + k_P \boldsymbol{\omega}_{mes}) \tag{213}$$

$$\dot{\hat{\boldsymbol{b}}} = -k_I \boldsymbol{\omega}_{mes} \tag{214}$$

where $\hat{b}$ is the estimated gyro noise, $\hat{v}_i$ are the reference vector in a known coordinate frame which have been rotated to body frame using the estimated attitude, v_i are the measured reference vectors in body frame and k_i, k_I and k_P are positive gains used to tune the observer. The attitude error is defined as

$$\hat{\boldsymbol{R}} := \hat{\boldsymbol{R}}_n^{bT} \boldsymbol{R}_n^b \tag{215}$$

which will converge towards $\boldsymbol{I}$.

4.4 Observer from Hua

The observer found in [17] has some similarities with the explicit complementary filter in [16]. It can be presented as [17]

$$\dot{\hat{\boldsymbol{v}}}^n = k_1(\boldsymbol{v}^n - \hat{\boldsymbol{v}}^n) + g\boldsymbol{e}_3 + \boldsymbol{Q}\boldsymbol{a}^b \tag{216}$$

$$\dot{\boldsymbol{Q}} = \boldsymbol{Q}S(\boldsymbol{\omega}_{n,b}^b) + k_v(\boldsymbol{v} - \hat{\boldsymbol{v}})\boldsymbol{a}^{bT} - \rho\boldsymbol{Q} \tag{217}$$

$$\dot{\hat{\boldsymbol{R}}} = \hat{\boldsymbol{R}}S(\boldsymbol{\omega}_{n,b}^b + \boldsymbol{\sigma}) \tag{218}$$

$$\boldsymbol{\sigma} = k_2 \boldsymbol{m}^b \times \hat{\boldsymbol{R}}_b^{nT} \boldsymbol{m}^n + k_3 \boldsymbol{a}^b \times \hat{\boldsymbol{R}}_{bT}^n(\boldsymbol{Q}\boldsymbol{a}^b + k_1(\boldsymbol{v} - \hat{\boldsymbol{v}})) \tag{219}$$

$$\rho = k_q \max(0, \|\boldsymbol{Q}\| - \sqrt{3}) \tag{220}$$

This observer uses measurements of $\boldsymbol{a}^b$ from an accelerometer, $\boldsymbol{\omega}_{n,b}^b$ from a rate gyro, $\boldsymbol{m}^b$ from a magnetometer and $\boldsymbol{v}^n$ from a gyro and estimates the attitude $\hat{\boldsymbol{R}}_b^n$ and the acceleration in NED frame $\dot{\hat{\boldsymbol{v}}}^n$.

Table 3: Geometric quantities found from manufacturer.

$\bar{c}$	0.214 m
b	1.68 m
S	0.355 m^2

5 Implementation

5.1 HINUAV-1

The UAV considered in this thesis is a SkyWalker EPO V2 airplane bought from bevrc.com. The airframe was assembled as a project by a former student Thomas Gansmoe. The total mass of the UAV was also measured to be 1.8 kg. The three geometric quantities were measured experimentally and then verified using documentation of the SkyWalker. The quantities can be seen in Table 3.

5.2 Hardware and data collection

The hardware on the UAV is based on a previous project at Narvik University College by a student named Benyamin Akdemir. The HINUAV-1 is equipped with four servos, one for each aileron, one for the rudder and one for the elevator. The airframe can be seen in Figure 6 the UAV is equipped with a Futaba 6EX 2.4GHz 6-channel transmitter and a Futaba R617FS receiver. The HINUAV-1 is equipped with a 3-axis accelerometer ADXL345 from sparkfun. It has a selectable measuring range from ±2g to ±16g and a resolution of 13 bit, which gives a precision of 4mg/LSB at the highest measuring range. The UAV is also equipped with a 3-axis gyroscope ITG-3000 from sparkfun. It has a measurement range of $\pm 2000^o$/sec with a resolution of 16-bits. This gives a precision of 14.375^o/sec/LSB. The gyro also has a temperature sensor to help compensate for temperature dependent drift. There is also a magnetometer onboard the UAV. It is also from sparkfun, the HMC5883L has an adjustable measuring range from ±0.88Ga to ±8.1Ga with a resolution of 12-bit, giving a precision between 0.73mGa to 4.35mGa depending on the measurement range. The magnetometer measures both the direction and magnitude of the magnetic field vector. The GPS onboard the UAV is an UP501 GPS receiver. It has a selectable refresh rate of 1Hz, 5Hz and 10Hz and a precision as high as 1.8 meter. It outputs data automatically in accordance with the standard NMEA-protocol. The messages can be chosen to be any combination of GGA, RMC, GSV and/or GSA. For this thesis only GGA messages were collected. Two pressure sensors are used on the HINUAV-1, there is one absolute pressure sensor MPX5100A that measure the absolute pressure and one differential pressure sensor MPX5010DP that is used with a pitot-tube. The pitot-tube can be used to estimate the

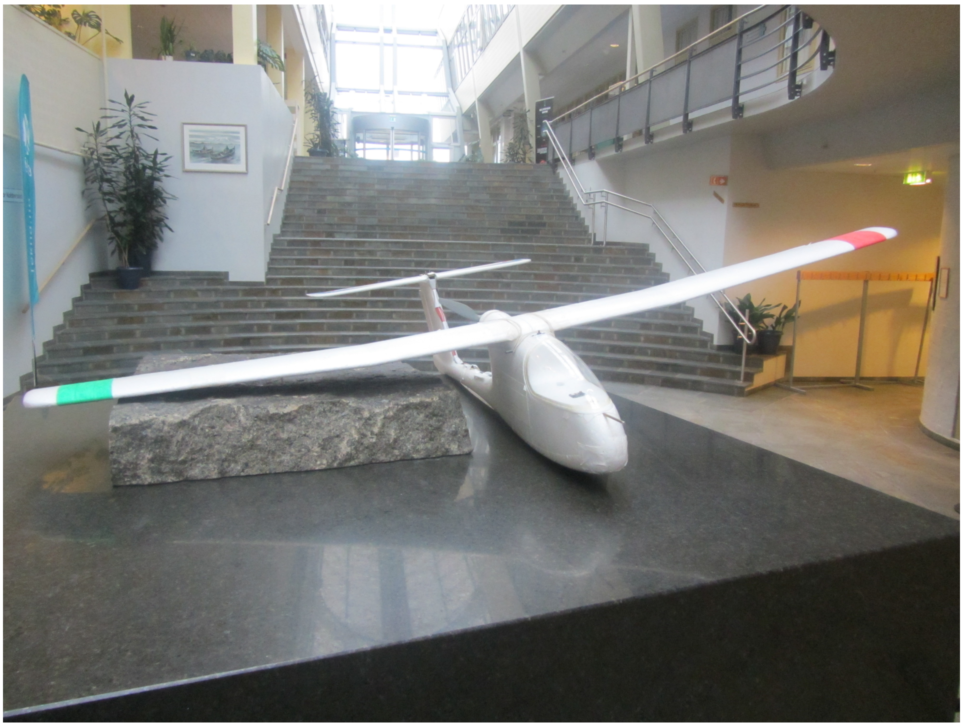

Figure 6: Picture of the HINUAV-1

wind direction and speed, while the absolute pressure sensor can be used to aid the altitude measurements of the GPS receiver.

These sensors make it possible to measure all the states necessary for aerodynamic estimation. The on-board processor is a FEZ Spider with a data logging algorithm written in C# based on the .NET Framework. The drivers for the sensors were initially written by Benyamin Akdemir, but modified to become more suitable for this thesis. Initially the hardware was not designed to capture control signals, but a work around was produced by setting up a second R617FS receiver which was connected to a MSP430F2416 microcontroller. The R16FS receiver generates control signals to the actuators which are pseudo pulse width modulated. The micrcontroller uses its capture peripherals to time the length of the pulses and sends it to the computer over UART where it is stored to a text file.

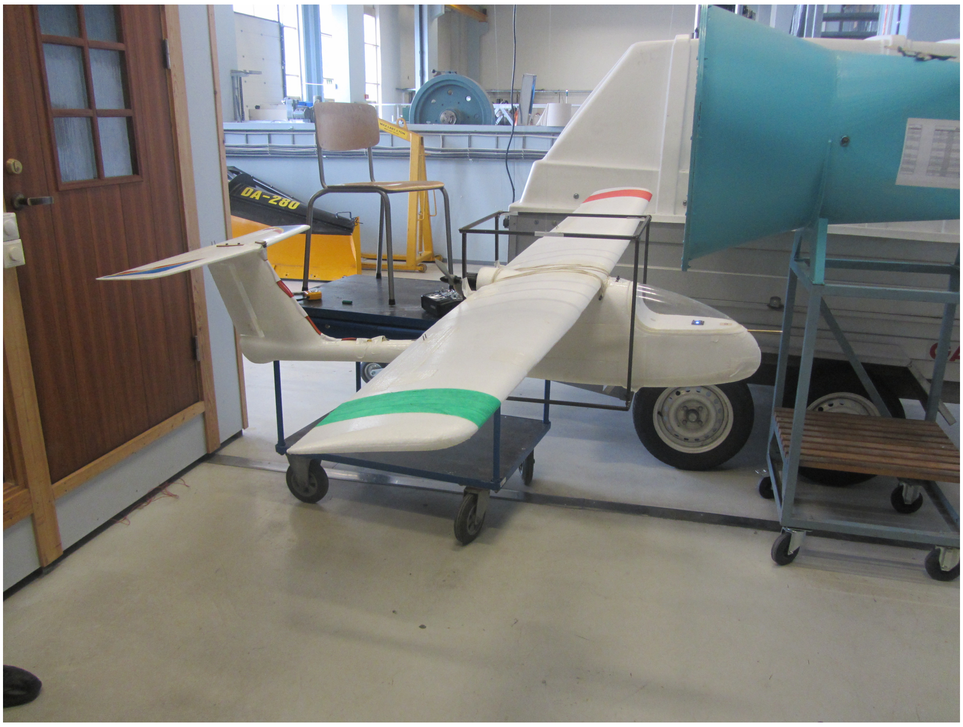

Figure 7: HINUAV-1 during trifilar inertia experiment, the Z-axis is being measured.

Table 4: Inertia values calculated from experiment.

I_{xx}	I_{yy}	I_{zz}
0.0827	0.147	0.169

6 Experimental Results

6.1 Inertia and geometry

The aircraft was suspended in a trifilar pendulum configuration as show in Figure 7. The measurements were done for each body axis of HINUAV-1 where the period of oscillation was recorded. The results obtained from the experiment can be seen in Table 4 where (88) was used to calculate the inertia values.

Figure 8: First flight of the HiNUAV-1

6.2 Flight data

A small series of test flights were performed on the 30th of June and data was successfully captured. The data captured can be seen in Figure 9-18. Figures 9 to 12 show the captured control signal. The value on y-axis is the pulsewidth of the pulse width modulated signal sent from the receiver to the actuators. The max pulse length is about 1.9-2ms and the shortest is about 1.0-1.1. As can be seen from the data of the Ailerons, Elevator and Rudder the data indicates that the control input was mostly in one direction for each control surface. For instance the elevator signal is mostly above 1.5 millisecond which indicates a lot of pitching downwards. This seems inconsistent with the GPS data.

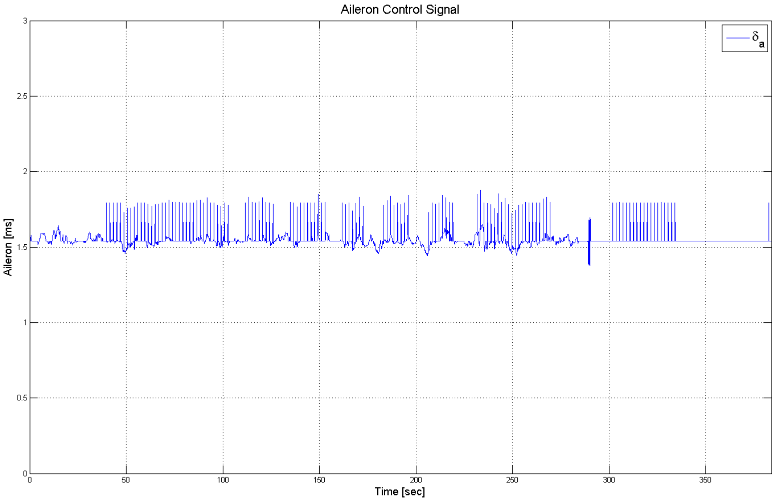

Figure 9: Aileron Control Signal

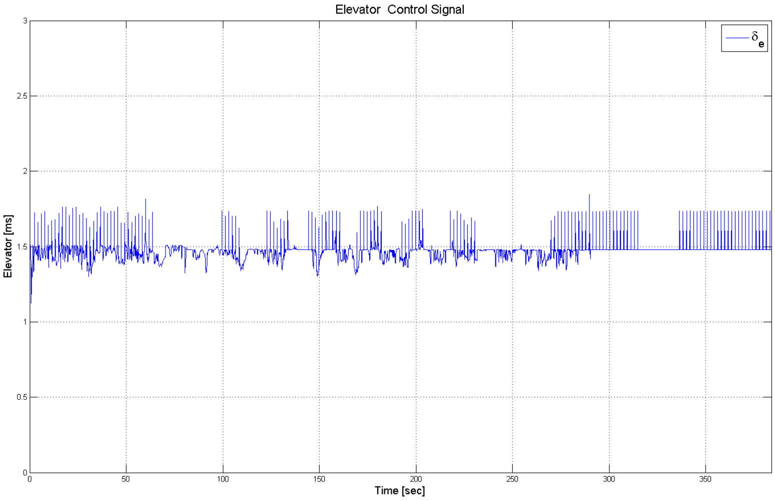

Figure 10: Elevator Control Signal

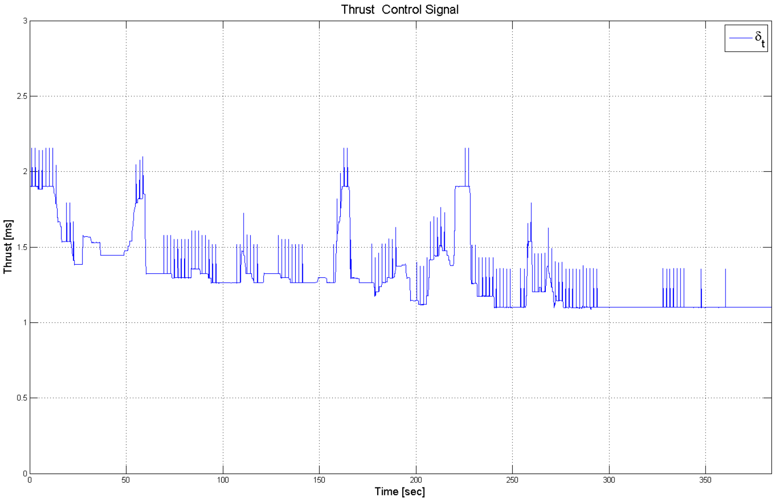

Figure 11: Thrust Control Signal

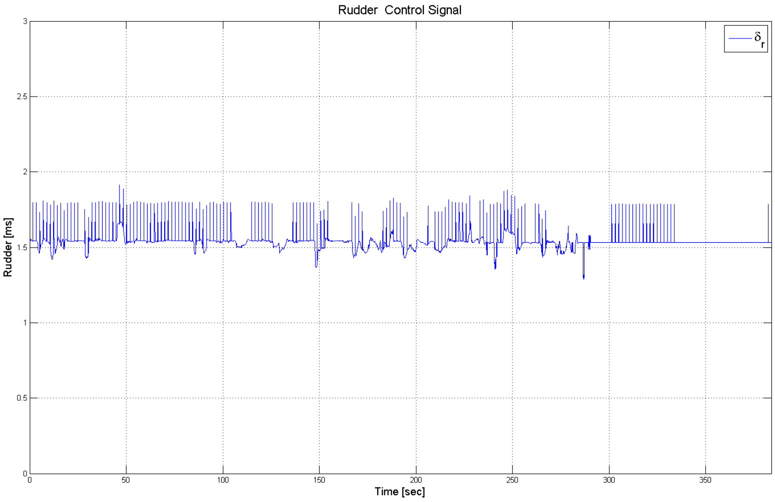

Figure 12: Rudder Control Signal

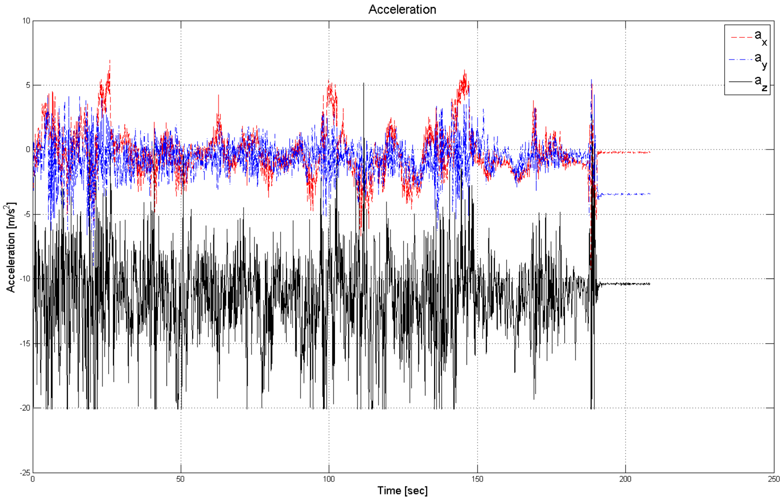

Figure 13: Accelerometer measurements

As can be seen from the sensor measurements there is a great deal of noise on the measurements. The noise of the accelerometer and gyro measurements as most likely from vibrations in the aircraft structure.

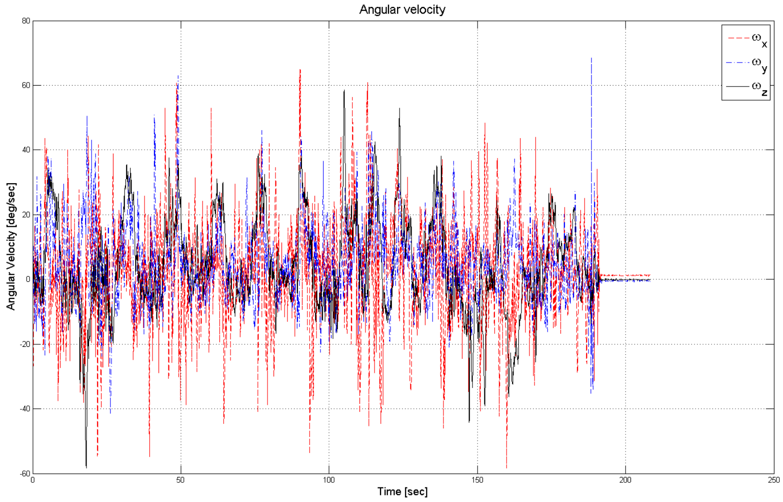

Figure 14: Gyroscope measurements

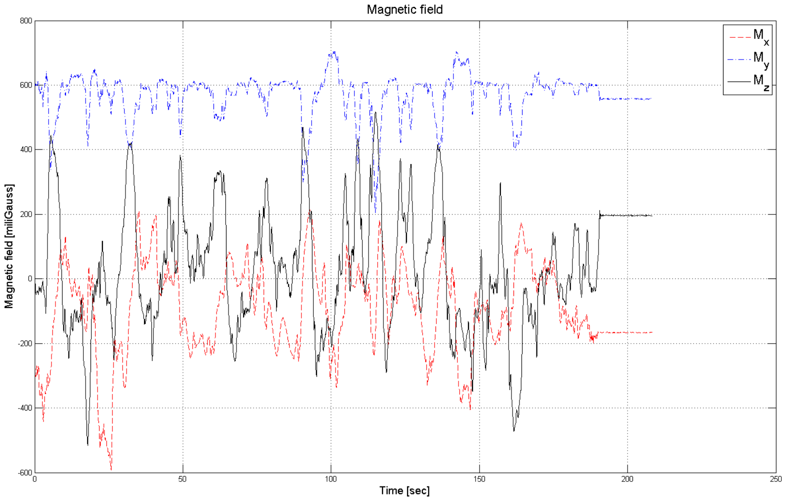

Figure 15: Magnetometer measurements

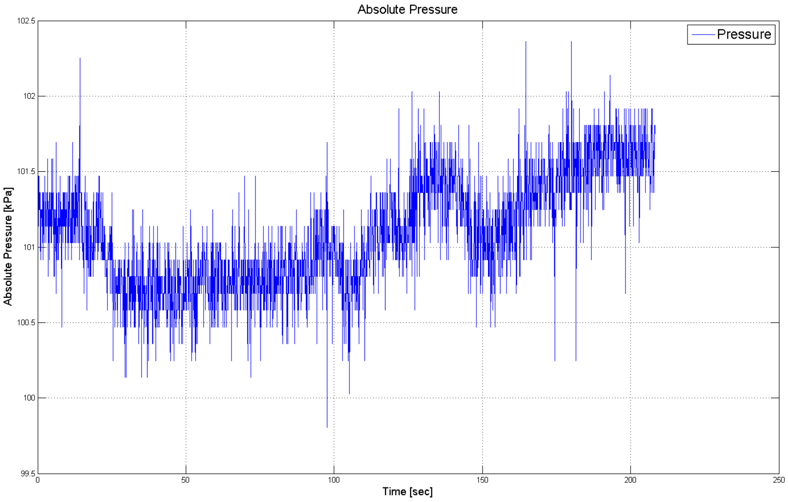

Figure 16: Absolute Pressure measurements

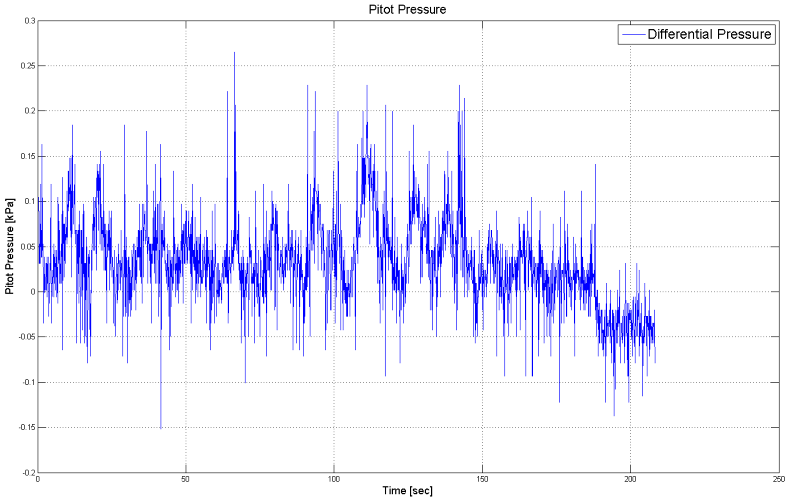

Figure 17: differential pressure measurements

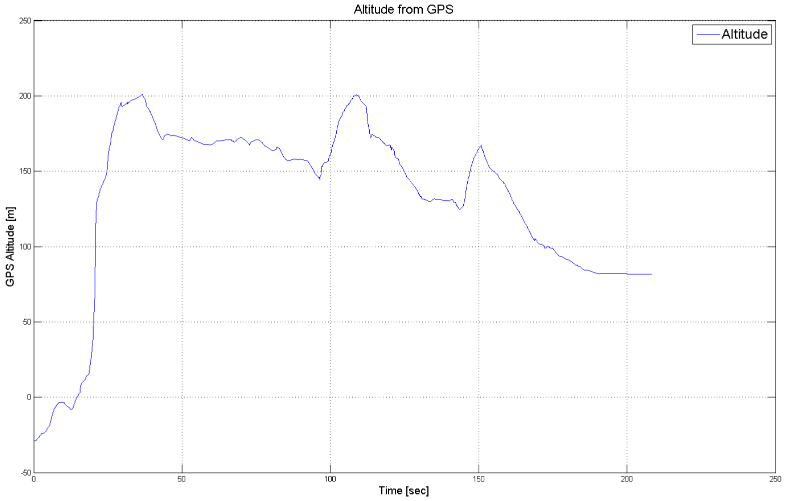

Figure 18: Altitude measurements from GPS

Figures 19 to 20 show the calculated position from GPS data of the UAV. The GPS signal has been decimated to match the number of samples from the other sensors. This will inherently introduce noise of the GPS measurements, but since GPS measurements have very low noise characteristic compared to the accelerometer and gyro, it should not cause any problems.

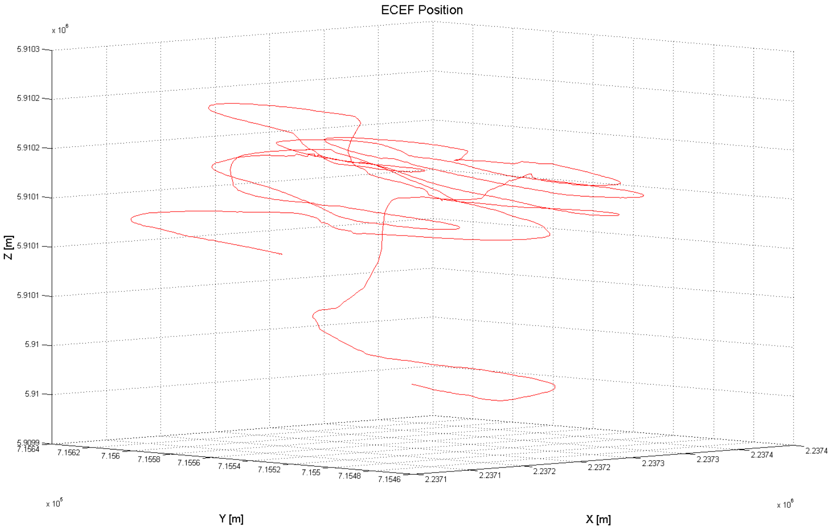

Figure 19: GPS position measurements in ECEF frame

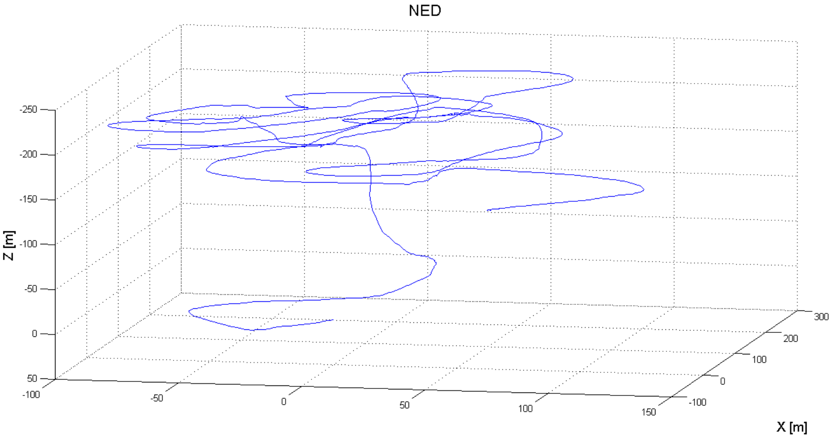

Figure 20: GPS position measurements in NED frame

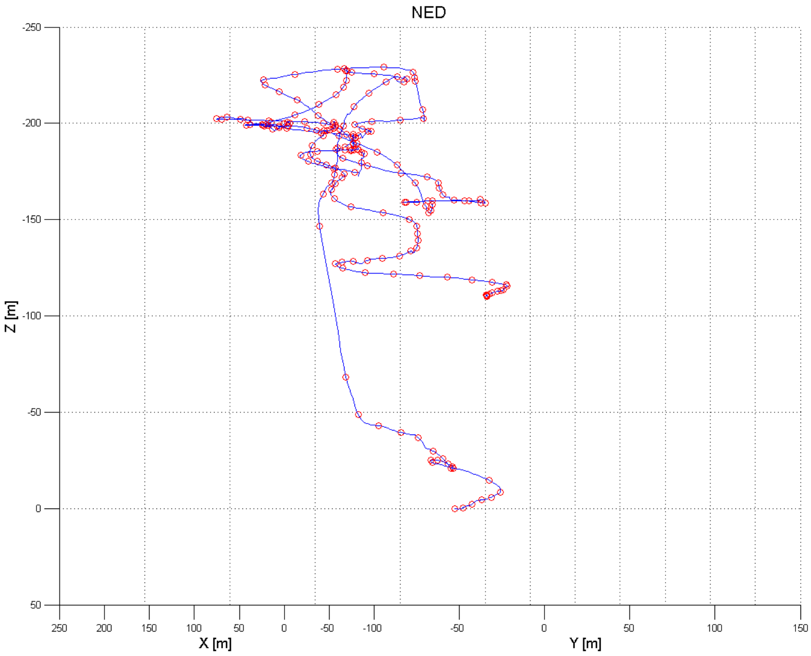

Figure 21: Position in NED frame with where each 10th measurement(1 measurement per second) is marked.

All the measurements start from the first sample in which the GPS had a fix on the location of the UAV. The GPS fix happened about 29 seconds into the flight. In Figure 21 a problem with the first GPS measurements is highlighted. From the graph where the space between 2 nodes represent 1 second of flight time it is clear the something happened in the beginning of the measurements. The UAV takes a turn and then starts climbing rapidly in altitude. From the graph it can be roughly estimated that in one second the UAV climbs from about -50 meters to almost 150 meters. This corresponds to a velocity of $360\frac{km}{h}$ which far outside the range of the UAV. Figure 22 show the magnitude of the differentiated position vector. As can be seen in the figure the is a sharp peek at about 30 seconds with a magnitude of over $500\frac{km}{h}$. It seems like the GPS wrongly estimated the altitude in the beginning and the after 30 seconds converged to the correct position. It is obvious that measurements should be take after this initial mistake in the altitude. The altitude calculated from the filtered pressure measurements does not show the same change in altitude as GPS does in the time interval.

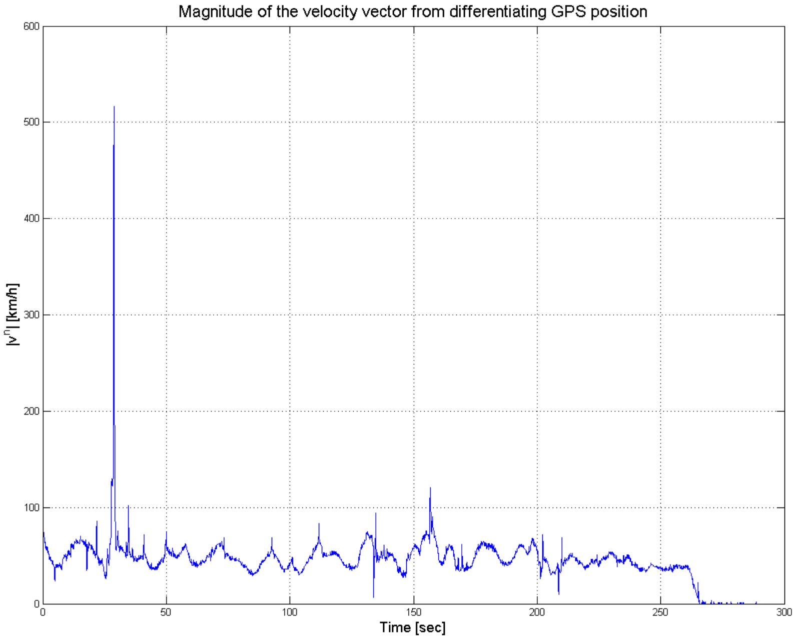

Figure 22: Magnitude of the differentiated GPS position vector before decimation.

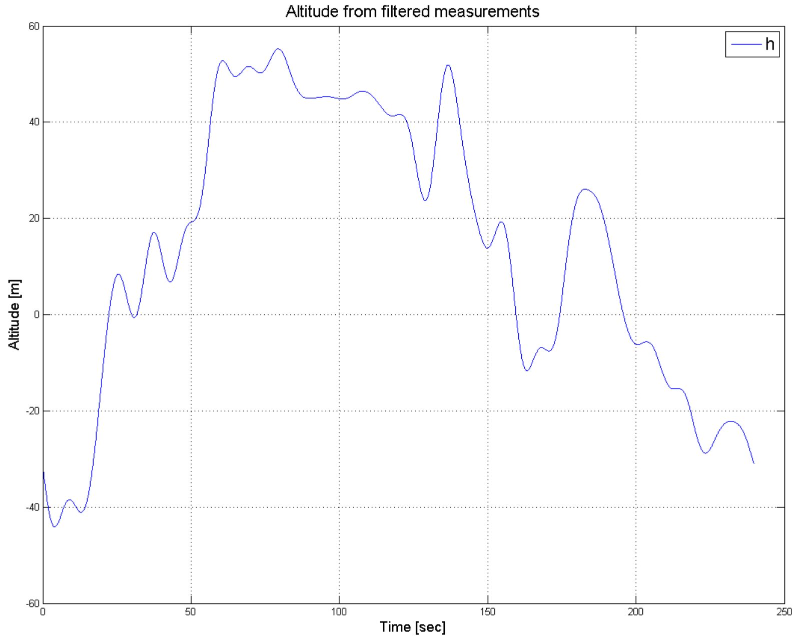

Figure 23: Altitude calculated from filtered absolute pressure measurements.

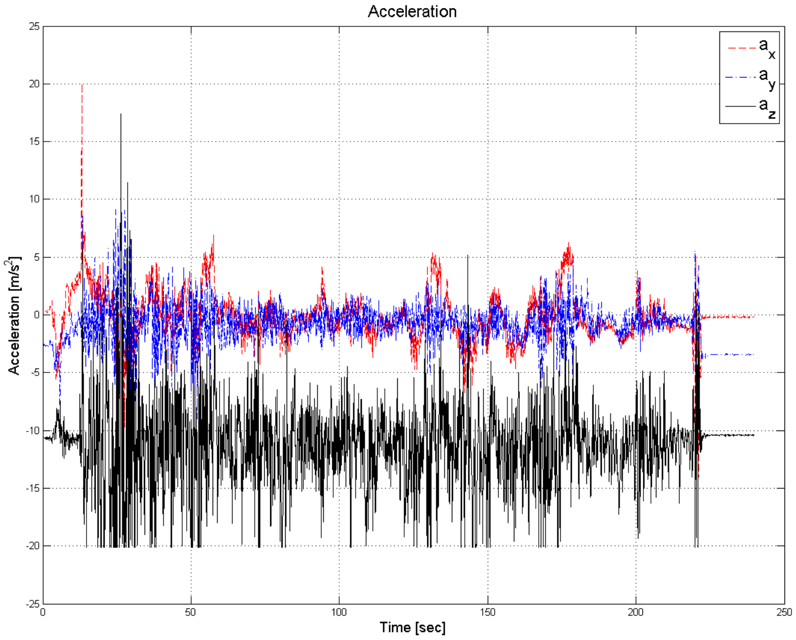

Figure 24: Accelerometer for the whole flight.

6.3 Filtered measurements

Since there was prevalent noise on the measurements it was difficult to infer the validity of the data. Therefore a simple low-pass filter based on fourier analysis was employed to remove some of the noise. Figures 25-40 show the comparison before and after filtering. Great care must be taken when filtering to not disrupt the signal. It would have been better to have implemented a wiener optimal filter which is more careful in removing higher frequency components.

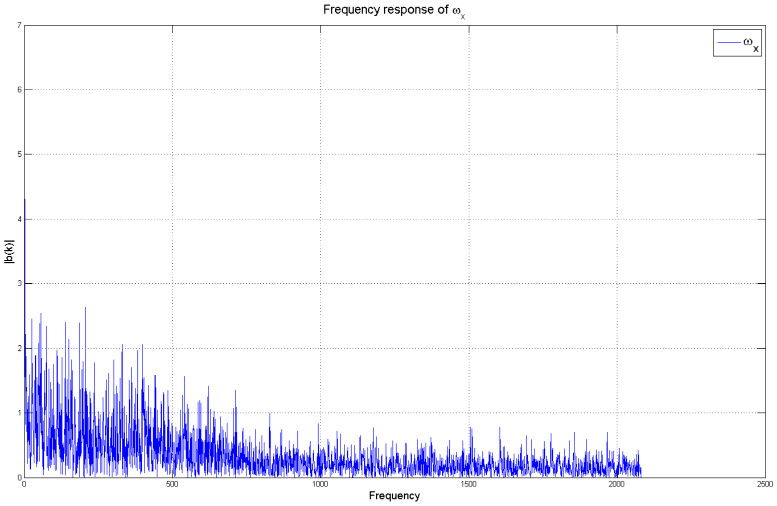

Figure 25: Frequency response of the measured angular velocity on x-axis

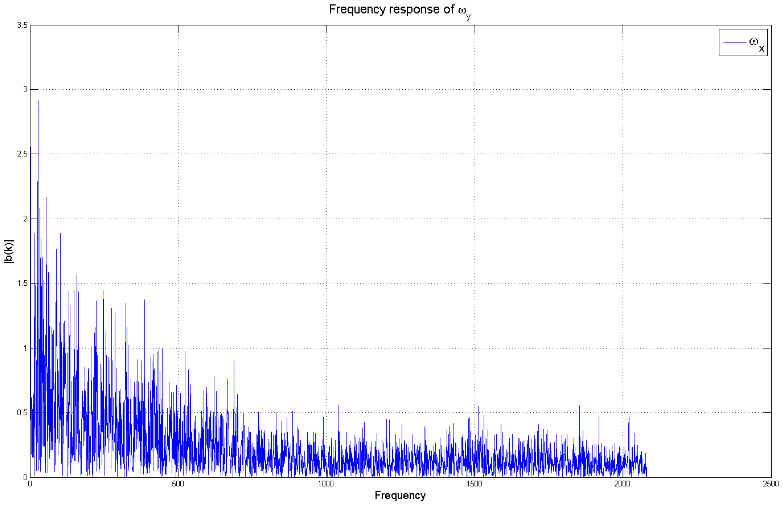

Figure 26: Frequency response of the measured angular velocity on y-axis

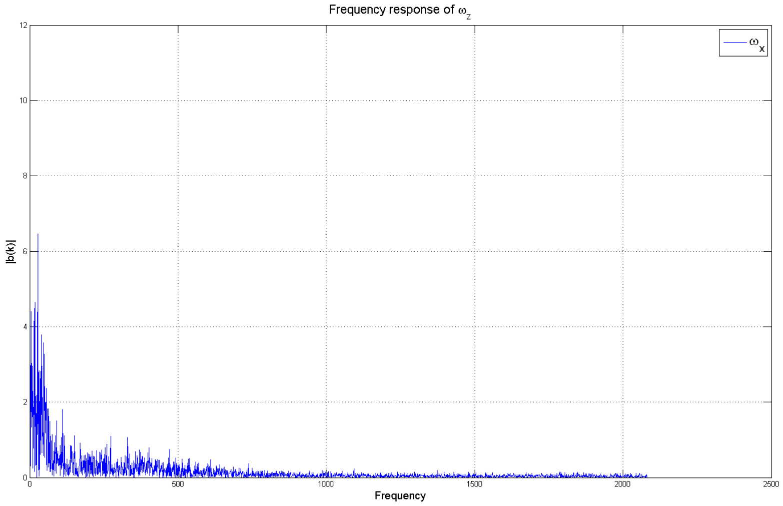

Figure 27: Frequency response of the measured angular velocity on z-axis

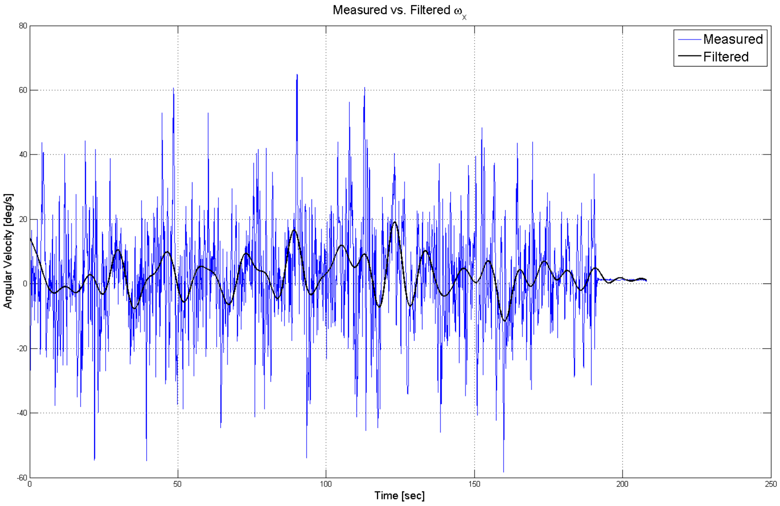

Figure 28: The measured angular velocity vs the filtered angular velocity on x-axis

In figure 25-27 the frequency plot for the angular velocity measurements are shown. The reconstructed angular velocity used a filter length of 50, which is a bit of a inbetween values. It filters away most of the noise, but it keeps the signal responsive enough to follow fast changes in the measured data.

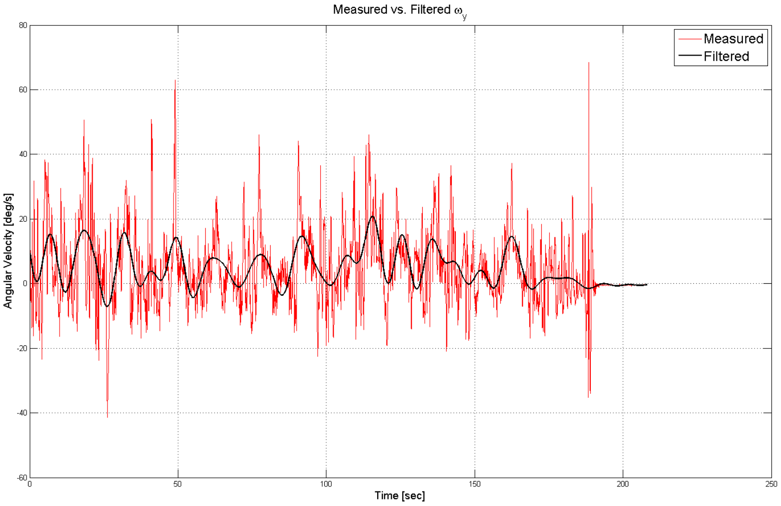

Figure 29: The measured angular velocity vs the filtered angular velocity on y-axis

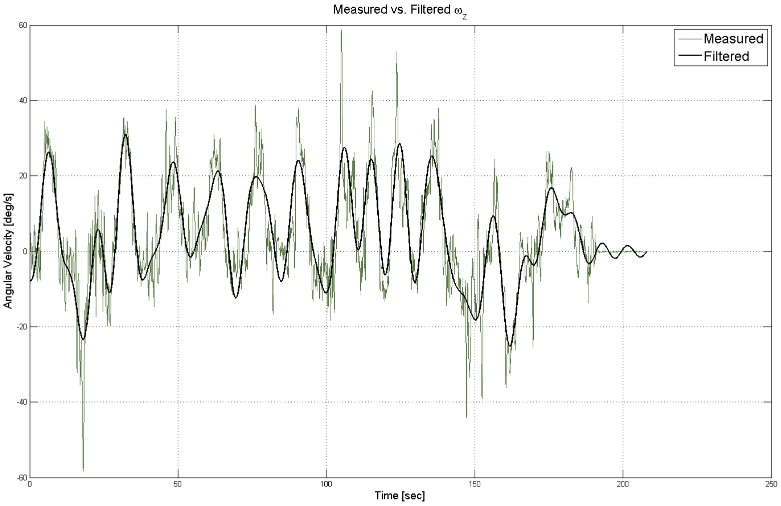

Figure 30: The measured angular velocity vs the filtered angular velocity on z-axis

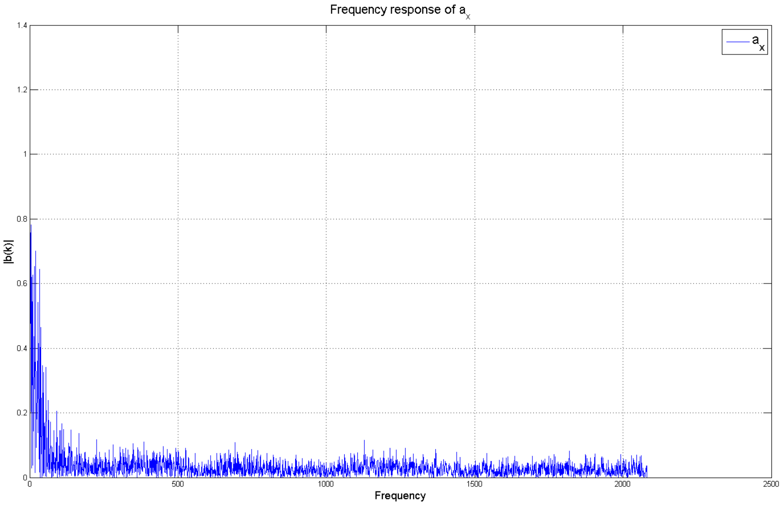

Figure 31: Frequency response of the measured acceleration on x-axis

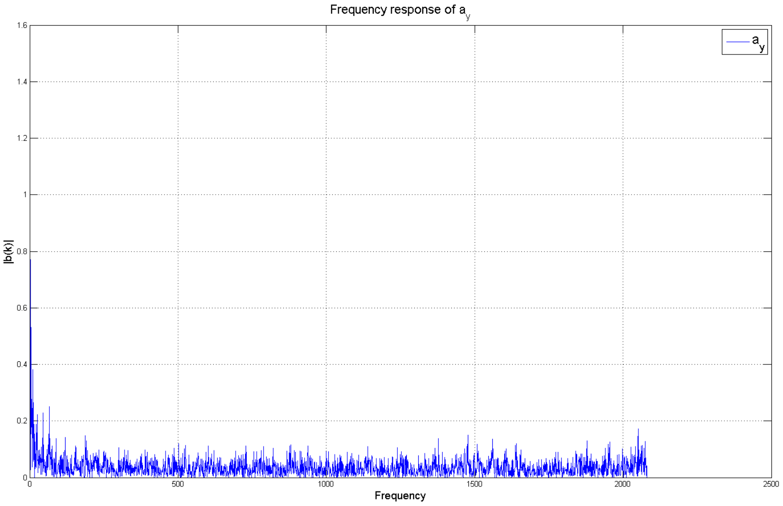

Figure 32: Frequency response of the measured acceleration on y-axis

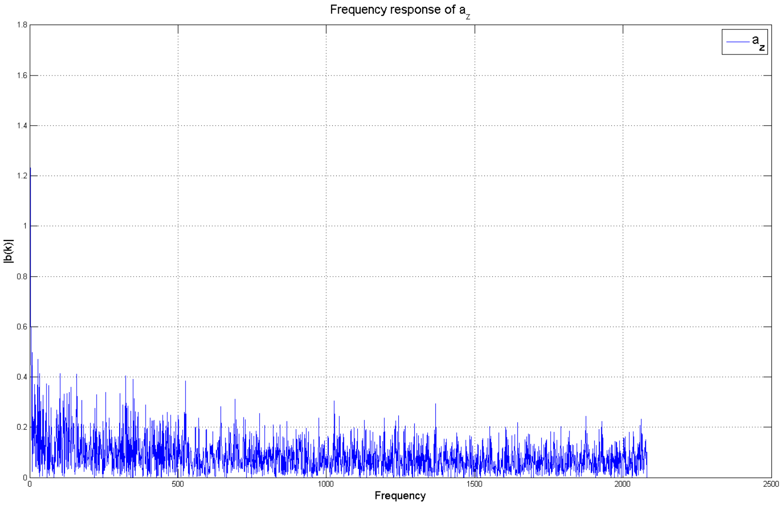

Figure 33: Frequency response of the measured acceleration on z-axis

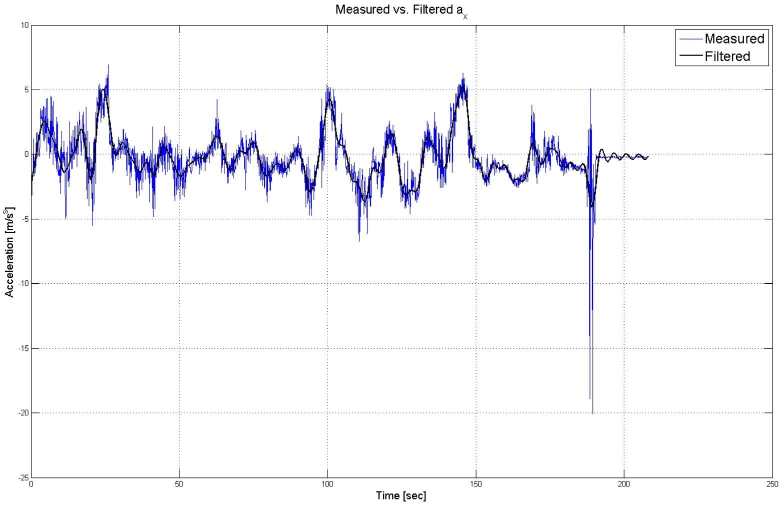

Figure 34: The measured acceleration vs the filtered acceleration on x-axis

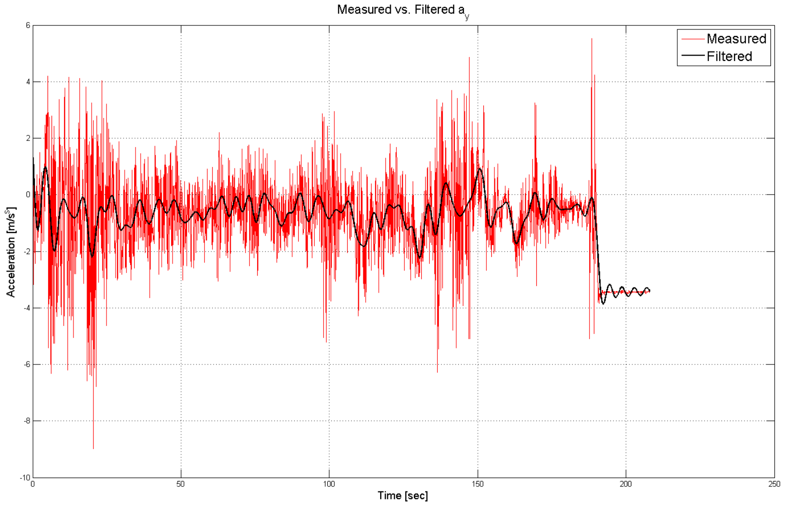

Figure 35: The measured acceleration vs the filtered acceleration on y-axis

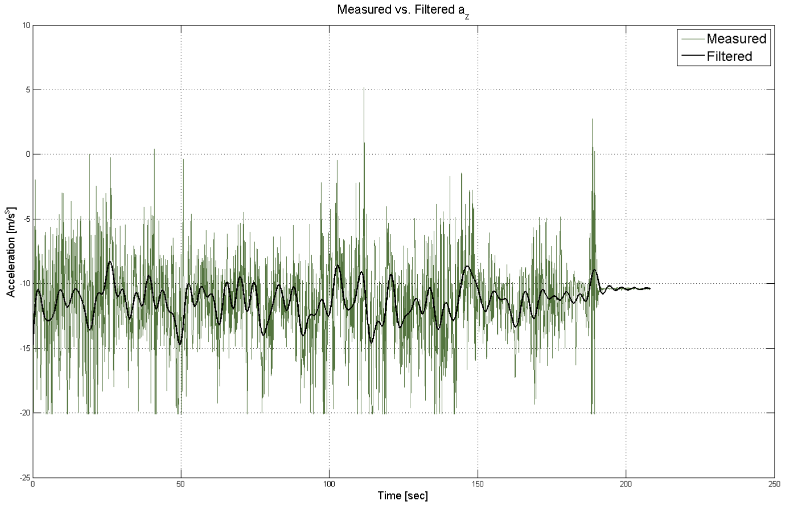

Figure 36: The measured acceleration vs the filtered acceleration on z-axis

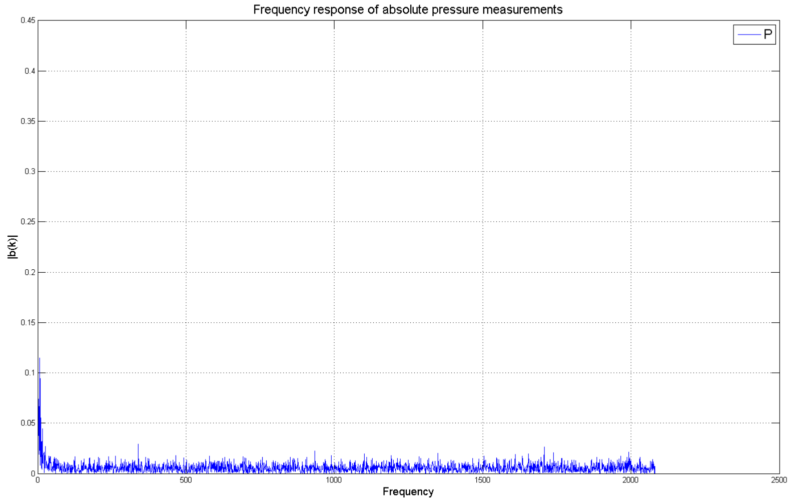

Figure 37: Frequency response of the measured absolute pressure

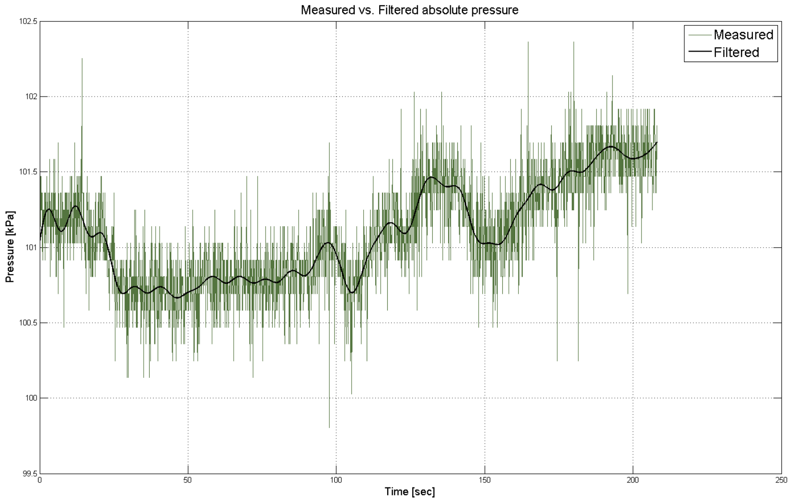

Figure 38: The measured absolute pressure vs the filtered absolute pressure

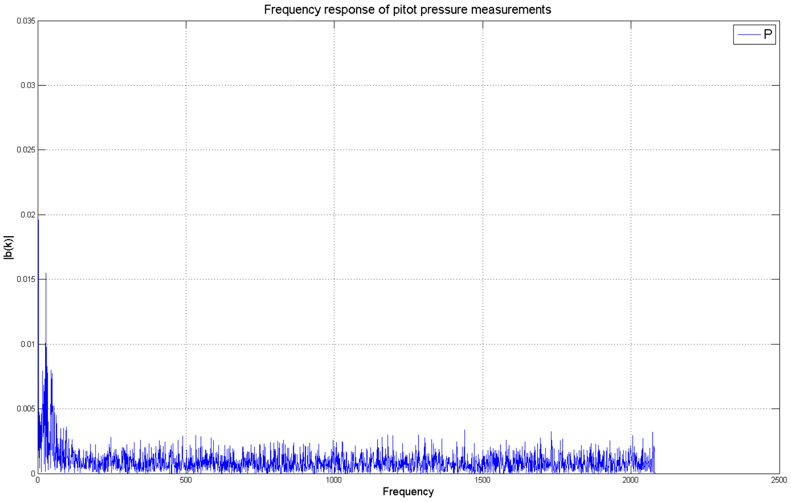

Figure 39: Frequency response of the measured pitot pressure

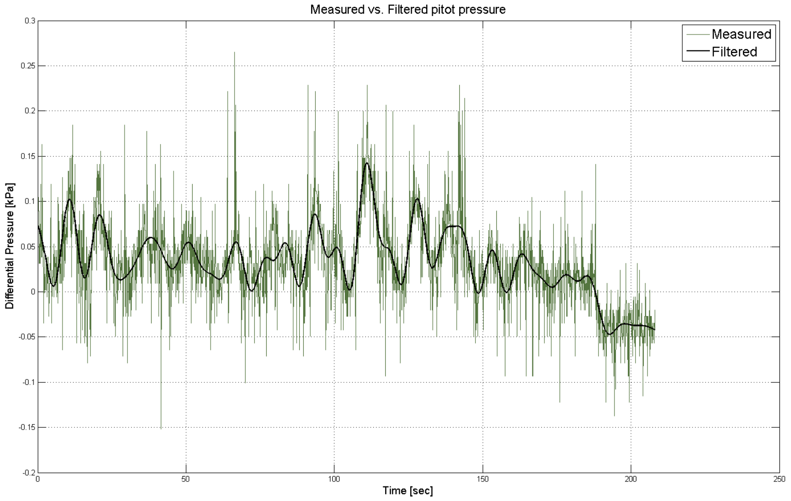

Figure 40: The measured pitot pressure vs the filtered pitot pressure

Table 5: Calculated magnetic vector based on IGRF11 model.

Dec	Inc	Horiz Intens	North Comp	East Comp	Vert Comp
6.99^o	77.56^o	11420.5 nT	11342.8 nT	1329.7 nT	51790.5 nT

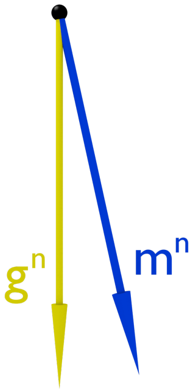

Figure 41: Illustration of the two reference vectors.

7 UAV state estimation Results

For the Explicit complementary filter the two known reference vectors were set to be

$$\boldsymbol{G}^n = \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} \tag{221}$$

and the magnetic field vector at the origin of the NED reference frame was calculated based on the calculator from NASA using the IGRF11 model. normalizing the vectors gives

$$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0.2139 \\ 0.0251 \\ 0.9765 \end{bmatrix} \tag{222}$$

A potential problem with the attitude estimation is highlighted in Figure 41. The two reference vectors almost point in the same direction which results in poor information about the attitude of the UAV. It is an early indication that the performance of the filter will be deterred. The Explicit Complimentary filter was tuned with the following gains values the initial

Table 6: tuning parameters for ECF

k_{acc}	k_{mag}	k_I	k_P
10	6	0.001	0.001

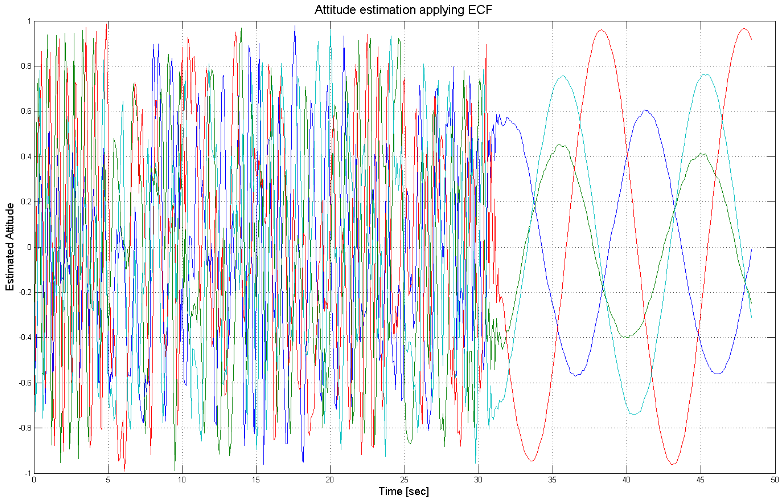

Figure 42: Attitude estimation using ECF with noisy measurements

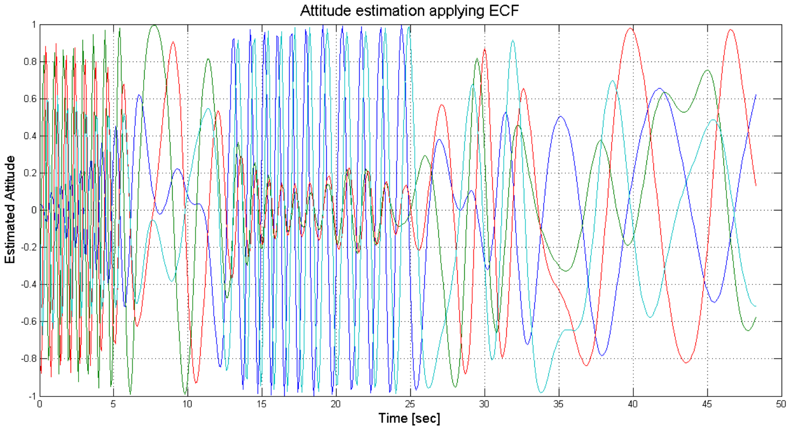

Figure 43: Attitude estimation using ECF with filtered measurements

value for the gyro bias was set as

$$\hat{\boldsymbol{b}}(0) = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \end{bmatrix} \tag{223}$$

for the initial attitude the TRIAD method was used to provide an estimate. Figure 42 show the result of the attitude estimation for the last 60 seconds of the flight. Its obvious from the figure that the filter did not converge, the reason for this might be the low amount of samples per second. The step time was set as 0.1 seconds since the measurements were taken 10 times per second. For the integration of the attitude and gyro bias the forward euler method was used which is known to give bad results for high step sizes. In Figure 43 filtered data was tried with the explicit complementary filter. The results suggest improvement, which is as expected although it still does not converge.

For the observer from [17], the velocity in NED frame is needed. Since the GPS only stored GGA messages the velocity has to be differentiated numerically from the position. Since the GPS data has been decimated the direct calculation of the velocity using (64) introduced noise to the velocity. This is nearly avoided when using (74). Figure 44 and 45 show the difference between the two differentiating methods. The Least Squares smoothing is equivalent to a low-pass filter with a cut-off frequency of 2Hz. Applying the observer from [17] did not yield desired results, as with ECF the observer did not converge. In Figure 46 and Figure 47 it can be seen that the filter is not operating as desired. The reason for this might be same as with the ECF filter i.e. low sampling of data.

There are two ways to increase the accuracy of numerical integration, the first is to increase the sampling rate of the data. In this case this is not possible without modification to the software logger and another flight test. The second method is to use a higher order approach, for instance the 4th order Runge-Kutta algorithm. By implementing RK4 in the Explicit complementary filter for integration of the attitude and using the trapezoid rule for integrating the bias. The result can be seen in Figure 48, immediatly it looks like an improvement from the previous results. However as can be seen from Figure 49-51, the measurements settle at around 30-40 seconds while the attitude estimation settles right after 15 seconds. Furthermore it can be seen from the magnetometer data and gyro data that they do not completely coincide.

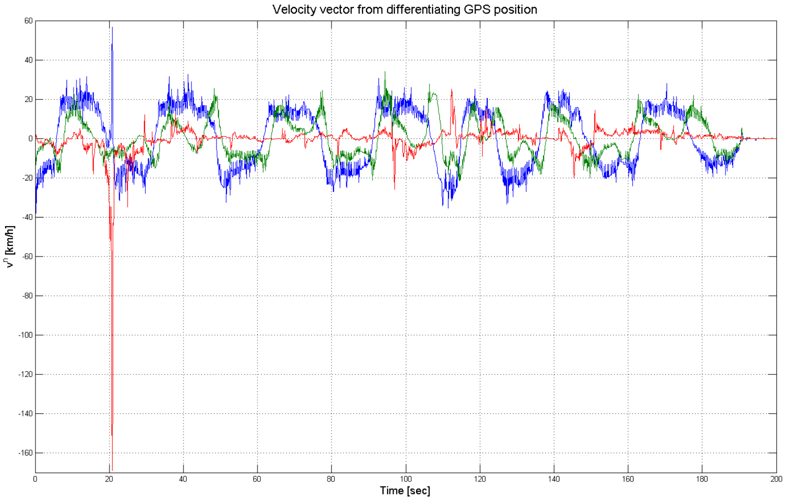

Figure 44: $\boldsymbol{v}^n$ obtained by using (64)

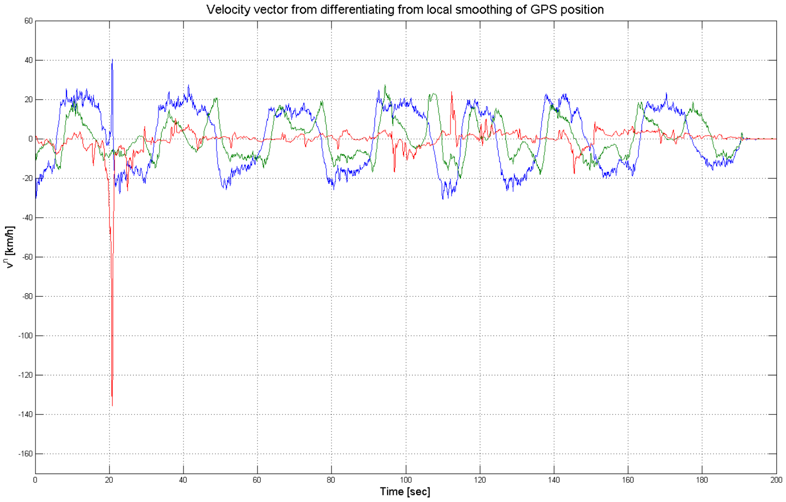

Figure 45: $\boldsymbol{v}^n$ obtained by using (74)

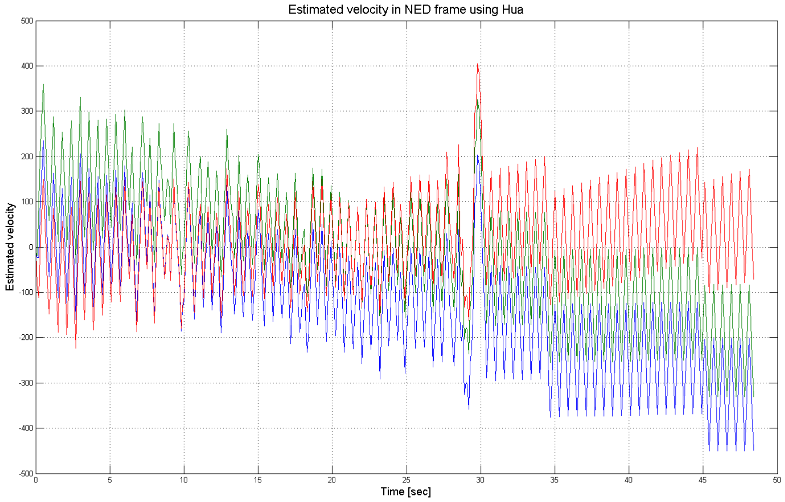

Figure 46: Estimated $\boldsymbol{v}^n$ using hua

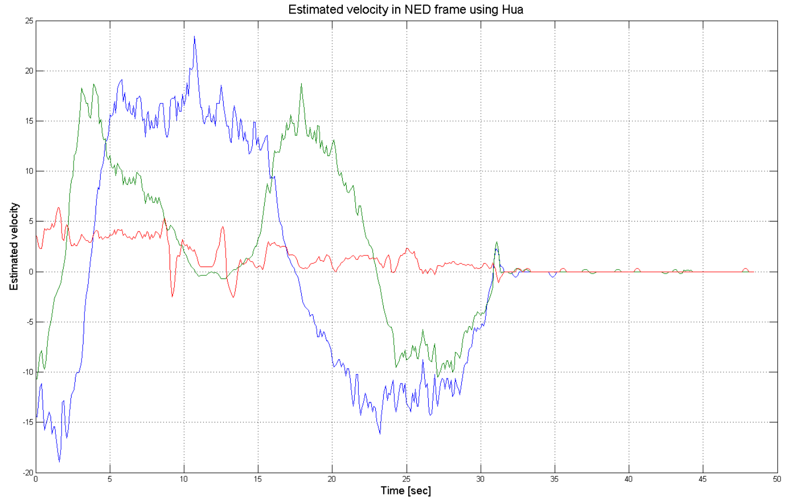

Figure 47: $\boldsymbol{v}^n$ obtained from differentiating $\boldsymbol{p}^n$

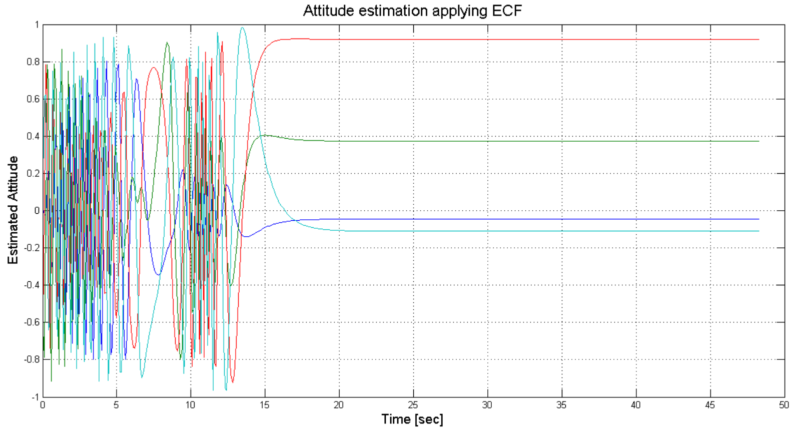

Figure 48: Estimated attitude using ECF with RK4 integration.

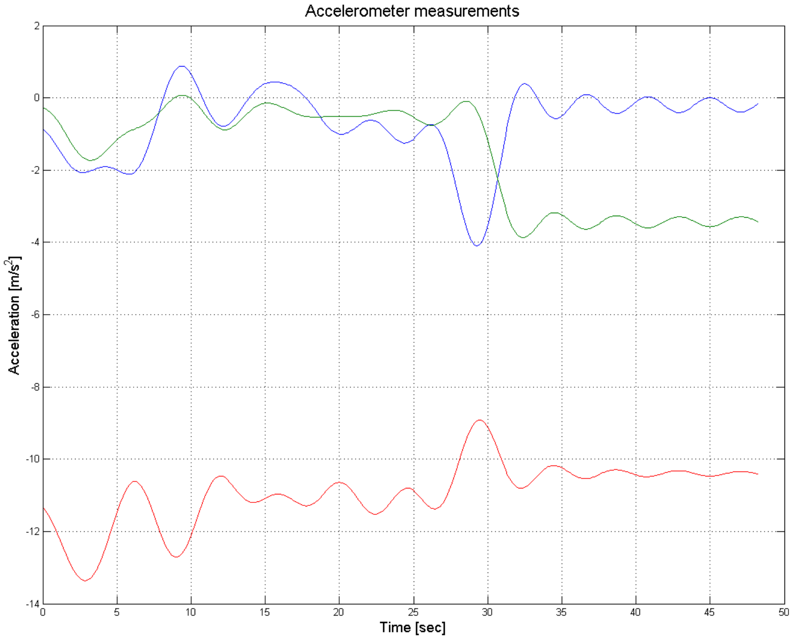

Figure 49: Accelerometer data used in ECF.

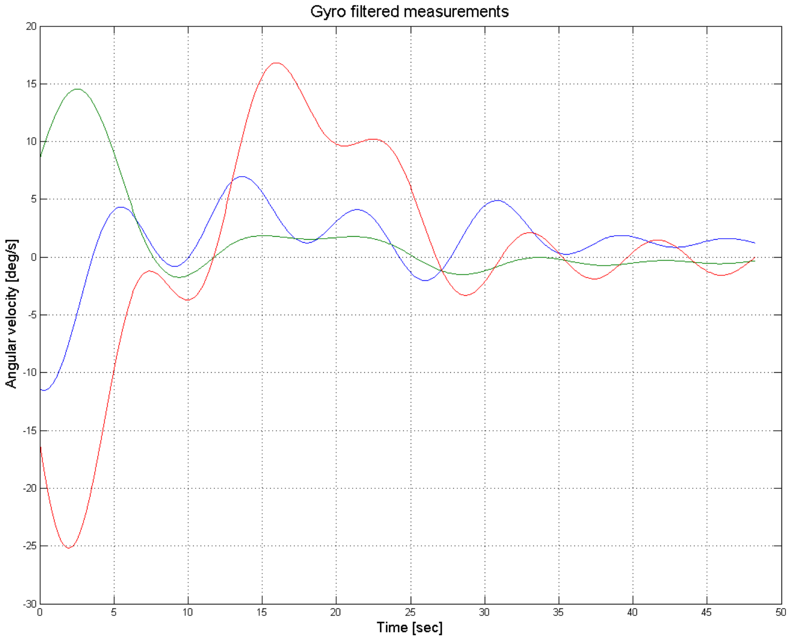

Figure 50: Gyro data used in ECF.

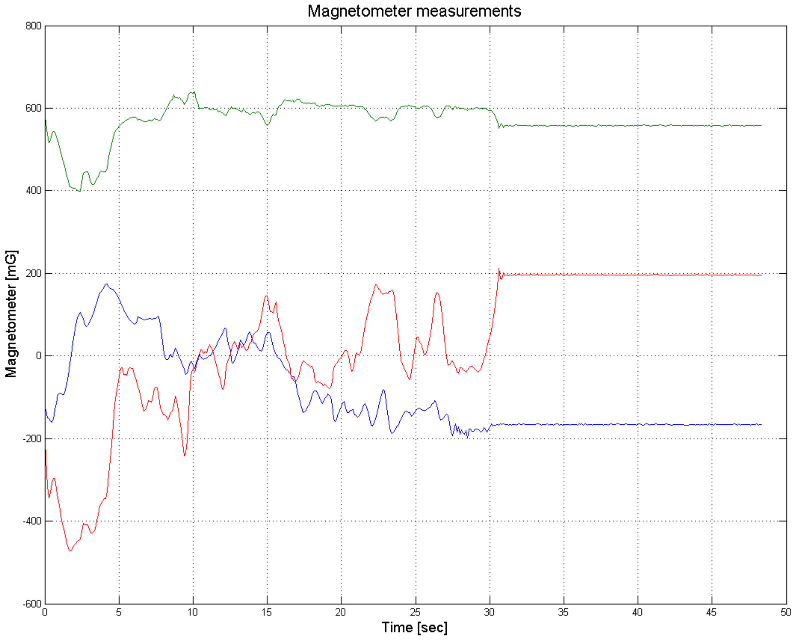

Figure 51: Magnetometer data used in ECF.

8 Aerodynamic model estimation Results

Since no states were estimated and the data from the flight was not ideal, the estimation of aerodynamic parameters was not performed. Since there was also difficulties with the relation between the logged control signals and the actual deflections of the control surfaces it would also be difficult to perform model validation of the estimated parameters.

9 Conclusion

In this thesis a framework for flight data gathering and analysis has been developed, in both hardware and software and used to collect flight data of a UAV. Two data loggers were written, one for logging control signals and one for logging the onboard sensor data onboard the HiNUAV-1. Theory of moment of inertia has been presented and implemented through experimental testing that was done to obtain the inertia matrix of the HINUAV-1. The result from the flight test was presented and has been discussed. Two non-linear observers were presented, but implementation was unsuccessful, it is likely that the sampling rate for the flight data was too low. The theory behind the Extended and the Linear Kalman filter was presented but not implemented because of problems with the state estimation. Methods for post processing data, such as filtering, smoothing and numerical differentiation, has been presented and applied to actual data with great success, giving further insight into the dynamics of the UAV. When collecting flight data it is important to perform data compatibility analysis, to verify that the measured data over the different sensors are consistent. Since there were uncertainties with the measured data, this analysis would have been difficult to perform and even more difficult to perform accurately. This thesis while lacking some results, will serve as a foundation for the future development of the HINUAV-1 parameter estimation. It is also important to note that when performing flight tests, it is important to have decided what the goal of the flight test is. What type of control inputs and maneuvers are desirable. Because of time constraints this was not possible and as a result the data used in the thesis were not ideal.

9.1 Future work

- The hardware onboard the HINUAV-1 should be done again with more careful planning, to ensure better performance of the sensors.
- To take full advantage of the FEZ Spider kit it is recommended to rewrite the software in native code, which is a lot faster and will potentially open up the possibility to increase the sampling rate by a significant amount.
- It can be seen from the accelerometer and gyro data that there is a lot of vibration in the structure of the HINUAV-1. To try and dampen this vibration reinforcement of the airframe might prove effective.
- In this thesis the capturing of the control signal was done through an extra receiver connected to a computer via UART, this can cause data to not be synchronized making the post-flight data processing more complicated. It would therefore be good to implement the control

signal logger onboard the UAV and have the capturing synchronized with the rest of the system.

- One specific problem is to try and relate the pulse-width modulated signal to the state of the actuator. Investigating this relationship will lead to accurate modeling of the thrust and control surfaces which will again improve the accuracy of the estimation.

- Some sensors have a wide measuring range, a lot wider than what is necessary, this leads to loss of accuracy and more noise on the data. Some sensors could be replaced with better alternatives. Some suggestions have been added to the digital appendix.

- When a good setup for logging data and a good aerodynamic model has been estimated, an accurate simulink representation of the UAV can the be constructed for preliminary testing of regulators and control algorithms.

References

[1] D. T. McRuer, D. Graham, and I. L. Ashkenas, *Aircraft Dynamics and Automatic Control*. Princeton University Press, 1973.

[2] R. F. Stengel, *Flight Mechanics*. Princeton University Press, 2004.

[3] T. P. Ehrhard, *Air Force UAVs: The Secret History*. Mitchell Institute Press, 2010.

[4] V. Strejc, "Least squares parameter estimation," in *Automatica*, vol. 16, 1980, p. 535 to 550.

[5] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Prentice Hall, 1999.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems," in *Journal of Basic Engineering*, 1960.

[7] L. A. McGee and S. F. Schmidt, "Discovery of the kalman filter as a practical tool for aerospace and industry," NASA, Tech. Rep., 1985.

[8] G. I. Smith, S. F. Schmidt, and L. A. McGee, "Application of statistical filter theory to the optimat estimation of position and velocity on board a circumlunar vehicle," NASA, Tech. Rep., 1962.

[9] L. Meng and L. Li, "Aerodynamic parameter estimation of an unmanned aerial vehicle based on extended kalman filter and its higher order approach," in *Advanced Computer Control*, 2010.

[10] G. Chowdhary and R. Jategaonkar, "Aerodynamic parameter estimation from flight data applying extended and unscented kalman filter," in *Aerospace Science and Technology*, 2010.

[11] L. Meng and L. Li, "Empirical aerodynamic modeling for robust control design of an oceanographic uninhabited aerial vehicle," in *International Conference on Electronics and Information Engineering*, 2010.

[12] A. Cho, J. Kim, and C. Kee, "Wind estimation and airspeed calibration using a uav with a single-antenna gps receiver and pitot tube," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 1, p. 109 to 117, January 2011.

[13] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB*. John Wiley & Sons, 2008.

[14] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," in *Proceedings of the IEEE*, 2004.

[15] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, “A new method for the nonlinear transformation of means and covariances in filters and estimators,” in *IEEE Transactions on Automatic Control*, 2000.

[16] R. Mahony, T. Hamel, and J.-M. Pflimlin, “Non-linear complementary filters on the special orthogonal group,” *IEEE Transactions on Automatic Control*, vol. 53, no. 5, p. 1203 to 1218, June 2008.

[17] M.-D. Hua, “Attitude estimation for accelerated vehicles using gps/ins measurements,” in *Control Engineering Practice*, 2010.

[18] T. Ingebretsen, “System identification of unmanned aerial vehicles,” Master’s thesis, Norwegian University of Science and Technology, 2012.

[19] M. J. Sidi, *Spacecraft Dynamics & Control: A Practical Engineering Approach.* Cambridge University Press, 2000.

[20] V. Klein and E. A. Morelli, *Aircraft System Identification: Theory and Practice.* AIAA, 2006.

[21] K. R. Britting, *Inertial Navigation Systems Analysis*, ser. Artech House GNSS Technology and Application. Artech House, 2010.

[22] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control.* John Wiley & Sons Ltd, 2011.

[23] K. Dutton, S. Thompson, and B. Barraclough, *Art of Control Engineering.* Prentice Hall, 1997.

[24] H. D. Curtis, *Orbital Mechanics for Engineering Students.* Elsevier Ltd., 2012.

[25] R. W. Beard and T. W. McLain, *Small Unmanned Aicraft: Theory and Practice.* Princeton University Press, 2012.

[26] D. McLean, *Automatic Flight Control Systems.* Prentice Hall, 1990.

[27] T. I. Fossen, “Mathematical models for control of aircraft and satellites,” Department of Engineering Cynernetics, Norwegian University of Science and Technology, Tech. Rep., 2013.

[28] E. A. Morelli, “Global nonlinear parametric modeling with application to f-16 aerodynamics,” in *Proceedings of the American Control Conference*, June 1998, p. 997 to1001.

[29] D. H. Titterton and J. Weston, *Strapdown Inertial Navigation Technology*, P. Zarchan, Ed. The Institution of Electrical Engineers, 2004.

[30] P. D. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems.* Artech House, 2008.

[31] S. Merhav, *Aerospace Sensor Systems and Applications.* Springer, 1996.

Table 7: DVD folder structure with explanation.

Folder Name	**Description**
Applied Space Technology	Previous work on UAV parameter estimation.
Control Signal Logger	Code implementation for capturing control signals.
Data Logger	Code implementation for collecting onboard sensor data.
Flight Data	The collected flight data used in this thesis.
Matlab	Various Matlab scripts used in this thesis.
PCB Designs	Schematics and Layouts of circuit boards designed during the thesis.
Pictures	Various pictures from the flight test and inertia experiment.
PropCalc_en_Win	Program for calculating thrust coefficient of propeller blades.
Simulink	Various simulink implementations, made during the thesis.

A Digital Attachment

On the DVD one will find the folder structure seen in Table 7