

State Estimation of an Airborne Wind Energy System

Master Thesis

to gain the academic degree

M.Sc.

at the Technical University of Munich (TUM),
Department of Electrical and Computer Engineering,
Institute for Electrical Drive Systems and Power Electronics

submitted by Martin Marmsoler, B.Sc.
Matriculation Number: 03652814

Contact: martin.marmsoler@tum.de, +49 1522 4675776

on June 17, 2019 in Munich.

1. Supervisor: Univ.-Prof. Dr.-Ing. Ralph Kennel
Institute for Electrical Drive Systems and Power Electronics

2. Supervisor: Florian Bauer, M.Sc.
Institute for Electrical Drive Systems and Power Electronics

Abstract

Airborne wind energy systems are a new technology which try to solve the problems of classic wind energy systems like high costs of material, installation costs and their limitation to shallow water near coasts due to their heights. The main challenges of airborne wind energy systems are the maximization of the retrieved wind energy and the automatic control of the kite because of high translational accelerations. In this thesis an attitude, position and velocity estimator was developed. This thesis proposes to use a Multiplicative Extended Kalman filter in inertia navigation system configuration with compensation of the sag of the tether. This filter type was validated with simulations in Matlab and also the effects when using different configurations and their drawbacks are described. The compensation of the tether sag is limited to the mass of the tether itself. No drag arised by the wind is compensated.

Statement

No sections of this thesis were submitted or used to gain another academic degree. Hereby I state, that this thesis was written alone by myself, except of explicitly marked passages.

Munich, June 17, 2019
Martin Marmsober

Acknowledgments

Hereby I would like to thank my supervisor M. Sc. Florian Bauer for his support during my work. Through him I got a great insight into filter design for kites for airborne wind energy systems with high dynamical movements, and he was able to answer my questions about the topic while writing this thesis.

Furthermore, I would like to thank my family and my friends for their advice and support.

Munich, June 17, 2019
Martin Marmsober

Contents

1	Introduction	1
2	Fundamentals	3
2.1	Coordinate systems	3
2.1.1	Earth fixed frame	3
2.1.2	Body fixed frame	3
2.1.3	Rotating from one frame to another	3
2.2	Sensors	7
2.2.1	Accelerometer	7
2.2.2	Gyroscope	8
2.2.3	Magnetometer	8
2.2.4	Global Positioning System (GPS)	11
2.2.5	Barometer	11
2.2.6	Tether angle sensor	11
2.3	Measurement systems	13
2.3.1	Attitude and heading reference system (AHRS)	13
2.3.2	Inertial navigation system (INS)	13
2.4	Filter	14
2.4.1	Kalman filter	14
2.4.2	Mahony explicit complementary filter	26
2.4.3	Comparing filter to already existing Attitude estimator Pixhawk 4	27
3	Simulations	29
3.1	Attitude and heading reference system (AHRS) filter	31
3.1.1	Euler angle based AHRS Extended Kalman filter (EKF)	31
3.1.2	Quaternion based AHRS EKF	34
3.1.3	AHRS multiplicative extended Kalman filter (MEKF)	34
3.1.4	Explicit complementary filter	36
3.2	Inertial Navigation System (INS) filter	36
3.3	Additional limitations	38
3.3.1	Loop time shorter than sensor reading	38
3.3.2	GPS module acceleration limitation	40
3.3.3	Tether sag	44
4	Preparation for practical use	49
4.1	Magnetometer calibration	49
4.2	C-Code generation	52
5	Conclusion and Outlook	53

List of Figures

1.1	Illustration of an airborne wind energy (AWE) system with the base station in the sea and the kite in the air.	2
2.1	Frame definition.	3
2.2	Definition of the Euler angles.	4
2.3	Magnetic inclination map.	9
2.4	Calculation of the position of the kite with the tether angle sensor.	12
2.5	Tether angle sensor with tether.	12
2.6	Block diagram of an AHRS, GPS and a Barometer to estimate all degrees of freedom.	13
2.7	State estimation with a full GPSINS.	14
2.8	Comparsion of a discrete Kalman filter with a time-continous Kalman-Bucy filter.	19
2.9	Blockdiagram of the Kalman-Bucy filter[34].	20
2.10	Blockdiagram of the discrete Kalman filter.	20
2.11	Filter dynamics comparsion.	21
2.12	Zoom in of figure 2.11.	22
2.13	Comparsion of the discretised Kalman filter with the discretised Kalman-Bucy filter at a sample frequency of 100 Hz.	23
2.14	Blockdiagram of the Mahony passive complementary filter.	26
2.15	Pixhawk 4 hardware.	28
3.1	Position and velocity of the kite.	29
3.2	Attitude of the kite.	30
3.3	Simulation result of the Euler AHRS EKF.	32
3.4	Estimated gyroscope bias. Due to the high translational acceleration, the gyro bias and the attitude have high errors.	33
3.5	Quaternion based AHRS EKF.	34
3.6	Attitude estimation of an AHRS MEKF.	35
3.7	Attitude estimation of the explicit complementary filter.	36
3.8	Estimated attitude.	37
3.9	Estimated gyroscope bias.	37
3.10	Estimated position and velocity.	38
3.11	Attitude Estimation with different sensor availability handling.	39
3.12	GPS Position with limitation to 4g.	41
3.13	Estimated angles of different filters and the error for the filter which does not calculate the correction step, with the GPS module limited to an acceleration of 4g and sample rates defined in section 3.3.1.	42
3.14	Position of INSMEKF DST, with the GPS module limited to an acceleration of 4g.	43
3.15	Tether sagging due to the mass of the tether.	45
3.16	Position and velocity of the INS MEKF filter without tether sag compensation.	47

List of Tables

2.1	Steps of the MEKF.	26
3.1	Sensor precision.	30
3.2	State precision.	30
3.3	Weights of the explicit complementary filter optimization.	36
3.4	Sample rates of different sensors.	38
3.5	Estimated angle error with different sensor availability handling.	39
3.6	Estimated position error with different sensor availability handling.	40
3.7	Estimated velocity error with different sensor availability handling.	40
3.8	Estimated angle error with 4g limitation of the GPS module.	40
3.9	Estimated position error with 4g limitation of the GPS module.	41
3.10	Estimated velocity error with 4g limitation of the GPS module.	41
3.11	Estimated angle error with and without tether sag compensation.	46
3.12	Estimated position error with and without tether sag compensation.	46
3.13	Estimated velocity error with and without tether sag compensation.	46
4.1	Test of the calibration of the magnetometer.	49

Nomenclature

Symbol	Description
x	$\in \mathbb{R}$, scalar variable
\boldsymbol{x}	$\in \mathbb{R}^n$, column vector
$\hat{\boldsymbol{x}}$	$\in \mathbb{R}^n$, column vector of estimated states
$\hat{\boldsymbol{R}}$	$\in \mathbb{R}^{n \times m}$, estimated matrix
\boldsymbol{X}	$\in \mathbb{R}^n$, vector of random variables
\boldsymbol{G}	$\in \mathbb{R}^{n \times m}$, matrix of size $n \times m$
I_x	unit matrix with size $x \times x$
$\boldsymbol{a} \times \boldsymbol{b}$	cross product of vector \boldsymbol{a} with vector \boldsymbol{b}
$\boldsymbol{a} \cdot \boldsymbol{b}$	scalar product of vector \boldsymbol{a} with vector \boldsymbol{b}
$\boldsymbol{\omega}_\times$	$\boldsymbol{\omega} \in \mathbb{R}^{3 \times 3}$, Cross product matrix
$vex(\boldsymbol{\omega}_\times) = \boldsymbol{\omega}$	Inverse of the cross product matrix
$\mathbb{P}_a(H) = \frac{1}{2}(H - H^T)$	anti-symmetric projection matrix of H
$\ \boldsymbol{x}\ $	euclidean norm
$\boldsymbol{p} \otimes \boldsymbol{q}$	quaternion multiplication
\boldsymbol{q}^{-1}	inverse of a quaternion
$\bar{\boldsymbol{q}}$	adjoint of a quaternion \boldsymbol{q}
$\ \boldsymbol{q}\ $	norm of a quaternion
\boldsymbol{g}_e	vector \boldsymbol{g} described in the earth frame
\boldsymbol{g}_b	vector \boldsymbol{g} described in the kite fixed body frame
\boldsymbol{m}_{mag}	vector \boldsymbol{m} described in the magnetic frame
R_e^b	rotation matrix to transform a vector in frame e to frame b
$\mathbf{Y}_{(n)}$	abbreviation of $\mathbf{Y}_n, \dots, \mathbf{Y}_1$
$E[X]$	mean of the random variable X
$\epsilon[T(X)]$	mean squared error of the estimator $T(X)$
$VAR(X)$	Variance of the random variable
$C_{\boldsymbol{X}_{n n-1}} = Cov(\boldsymbol{X}_n, \boldsymbol{X}_n \mathbf{Y}_{n-1})$	Conditional covariance matrix of the random variable \boldsymbol{X} at step n conditioned by \mathbf{Y}_{n-1}
$f_\Theta(\theta; \sigma)$	Probability density function (PDF) of the random variable Θ with mean θ and variance σ
$f_{X \Theta}(x \theta)$	PDF of X with the knowledge of Θ
$\hat{\boldsymbol{x}}_{n n-1}$	estimated state at step n with the measurements from step $n-1$
$\hat{\boldsymbol{x}}_{n n}$	estimated state at step n with the measurements from step n

List of Symbols

α	error angle used in the MEKF	—
η_{ω_b}	white gaussian noise from the gyroscope	
η_{a_b}	$\in \mathbb{R}^3$, gaussian noise of the accelerometer in body coordinates	m s^{-2}
$\eta_{P_{GPS}}$	$\in \mathbb{R}^3$, position noise from the GPS module	m s^{-1}
$\eta_{v_{GPS}}$	$\in \mathbb{R}^3$, velocity noise of the GPS module	m s^{-1}
γ	$\in \mathbb{R}^3 [\phi \quad \theta \quad \psi]^T$, attitude of the kite	rad
ω_b	$\in \mathbb{R}^3$, angular rate in the body frame	rad s^{-1}
ω_b	$\in \mathbb{R}^3$, angular rate in the inertia frame	rad s^{-1}
a_b	$\in \mathbb{R}^3$, acceleration in the body coordinates	m s^{-2}
a_{cp}	$\in \mathbb{R}^3$, centripetal acceleration	m s^{-2}
b_ω	$\in \mathbb{R}^3$, constant/ slow varying gyroscope bias	rad s^{-1}
g_b	$\in \mathbb{R}^3$, gravity vector in body coordinates	m s^{-2}
g_e	$\in \mathbb{R}^3$ the gravity vector in the inertia frame	m s^{-2}
m_e	$\in \mathbb{R}^3$, magnetic field in inertia coordinates	μT
m_{mag}	$\in \mathbb{R}^3$, magnetic field in magnetic coordinates in the direction of the magnetic north pole	μT
$p_{GPS_{meas}}$	$\in \mathbb{R}^3$, GPS position measured from the GPS module	m
p_{GPS}	$\in \mathbb{R}^3$, GPS position relative to the base station	m
v_b	$\in \mathbb{R}^3$, velocity in the body coordinates	m s^{-1}
v_e	$\in \mathbb{R}^3$, velocity in inertia frame	m s^{-1}
v_{GPS}	$\in \mathbb{R}^3$, velocity in inertia coordinates	m s^{-1}
Δt	sample time	s
δ_d	magnetic declination	°
δ_i	magnetic inclination	°
\dot{v}_b	$\in \mathbb{R}^3$, translational acceleration in the body coordinates frame	m s^{-2}
η_p	white gaussian noise of the measured pressure	N m^{-2}

η_r	gaussian white noise of the tether length measurement	m
$\eta_{\phi_{base}}$	gaussian white noise of the elevation angle measurement in the base station	rad
$\eta_{\phi_{kite}}$	gaussian white noise of the elevation angle measurement in the kite	rad
$\eta_{\theta_{base}}$	gaussian white noise of the azimuth angle measurement in the base station	rad
$\eta_{\theta_{kite}}$	gaussian white noise of the azimuth angle measurement in the kite	rad
ϕ	rotation angle about the x axis (roll)	rad
ϕ_{base}	elevation angle measured with the tether angle sensor in the base station	rad
ϕ_{kite}	elevation angle measured with the line angle sensor in the kite	rad
ψ	rotation angle about the z axis (yaw)	rad
θ	rotation angle about the y axis (pitch)	rad
θ_{base}	azimut angle measured with the tether angle sensor in the base station	rad
θ_{kite}	azimut angle measured with the line angle sensor in the kite	rad
B	strength of the magnetic field	µT
h	absolute height	m
p	air pressure at height h	$N m^{-2}$
p_0	air pressure at sea level	$N m^{-2}$
p_{tether}	position determined from the tether angle sensor	m
r	tether length	m
v_{TAS}	True airspeed	$m s^{-1}$
INSMEKF DST	INS filter based on the MEKF which does not calculate the correction step for the non available filter	

List of Abbreviations and Terms

AHRS Attitude and heading reference system

AWE airborne wind energy

AWES airborne wind energy system

EKF Extended Kalman filter

GPS Global Positioning System

IMU Inertial Measurement Unit

INS Inertial Navigation System

MEKF multiplicative extended Kalman filter

MSE mean squared error

NED North-East-Down

PCB Printed Circuit Board

PDF Probability density function

RMSD root-mean-square deviation

Chapter 1

Introduction

Every year the energy produced by renewable systems increases, and in 2018 this type of energy production in Germany exceeded 40 % of all produced energy[1]. In 2018 the largest part of renewable energy was produced by wind turbines with about 26 %[2], but wind turbines have the disadvantage of being tall and having a high weight with high offshore installation costs. No commercial floating wind turbine was built before this work was written. Due to their height, large movements are induced by waves[2] and shear forces from the wind [3], so a floating wind turbine is a challenge. A new not commercially used technology called airborne wind energy system (AWES) is developed to solve some key issues of a wind turbine. The AWES is illustrated in figure 1.1. It tries to solve some key issues of a wind turbine. An AWES consists of a kite which is connected by a tether with a base station and generates electrical power by flying crosswind[4]. Due to the increasing wind speed, which results in increasing kinetic wind energy with gaining altitude the AWESs operates in higher altitudes in order to generate more energy. While gaining altitude for a wind turbine has the disadvantage, needing much more material and costs for installation, for an AWES only the length of the tether and the size of the kite change, but the height of the base station increases minimal. Another benefit of an AWES is, that the base station is much smaller and not as tall as the tower of a wind turbine. So building floating base stations is much easier and less expensive. To generate the maximum power the kite flies fast crosswind trajectories, such as eight-figures or circles[5]. In these cases the measured onboard acceleration is corrupted by centripetal and translational acceleration whereas measuring the acceleration and estimating the attitude is not possible anymore[6]. A second problem is due to the high accelerations the GPS module is limited to, and so making these measurements not always available. But there are sensors like a tether angle sensor on both sides, at the kite side and at the ground station side, which can be used to determine the position next to the GPS module.

In chapter 2 of this thesis the fundamentals about the used sensors and filters are described. In chapter 3 multiple filters like the EKF or the Mahony explicit complementary filter are tested in different configuration. Using tether angle sensors to estimate the position of the kite was already made from Sedelmaier in [6], but with the assumption the tether is a straight line. In this thesis an approach is presented to estimate the position of the kite with compensating tether sag induced by the tether's mass. This compensation is implemented into the filter and compared with the filters, which do not respect the tether sag. From the most promising filter C-Code is generated, which is described in chapter 4. Due to the interference of the environment with the earth magnetic field, an approach to calibrate the magnetometer is presented and compared to the uncalibrated one which is presented also in chapter 4.



Source: [7]

Figure 1.1: Illustration of an AWE system with the base station in the sea and the kite in the air.

Chapter 2

Fundamentals

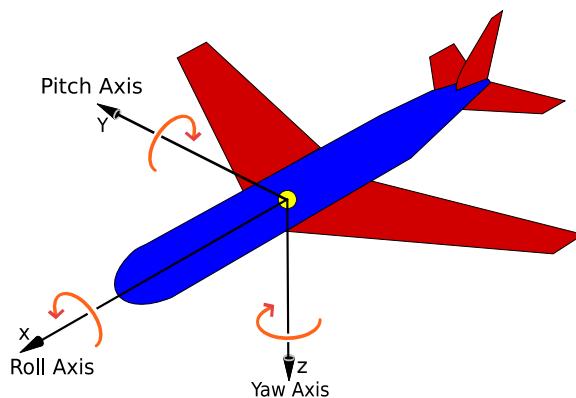
2.1 Coordinate systems

2.1.1 Earth fixed frame

To determine the position and the attitude of the kite, a fixed frame, which does not move or rotate with the kite is defined. This frame is tangential to the base station and the x-Axis directs to geographic north, the y-Axis to east and the z-Axis downward. It is called North-East-Down (NED) frame[6, 8].

2.1.2 Body fixed frame

The accelerometer, the magnetometer and the gyroscope are mounted on the kite, whereas the measured data changes with the attitude of the kite. For this a new frame which is fixed to the body of the kite is introduced as shown in figure 2.1. The x-Axis is aligned with the nose of the kite and always directs in the direction of flight. The y-Axis shows to the right from the x-Axis, and the z-Axis downward to get a right handside orthogonal system. The rotations are defined as mathematically positive rotations around the corresponding axis. In this definition, a positive angle around the y-Axis named pitch angle means the angle of attack increases.



Source: Adapted from [9]

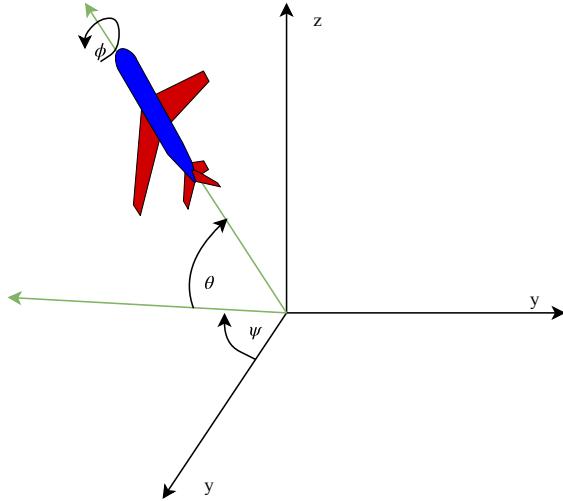
Figure 2.1: Frame definition.

2.1.3 Rotating from one frame to another

To determine the sensor data in the earth fixed frame or determining the gravity or the magnetic field in the body frame, rotations must be defined.

Euler angles

A common way to determine the attitude of a body in a different frame are the Euler angles. This representation is more representative than others like the quaternions described in section 2.1.3. The drawback is, that in specific angles, like the pitch angle at $\pm 90^\circ$, the calculations between different frames have singularities. In these cases it is impossible to determine the angles uniquely, also known as gimbal lock[10]. A second disadvantage is the need of higher computational power, due to the trigonometric functions. The calculation is slower than calculating with quaternions[10]. The Euler angles are defined from the earth fixed frame to the body fixed frame shown in figure 2.2[10].



Source: Adapted from [9]

Figure 2.2: Definition of the Euler angles.

Rotating a vector \mathbf{z} from the earth fixed frame to the body fixed frame is defined as

$$\mathbf{z}_b = \mathbf{R}_e^b \mathbf{z}_e \quad (2.1)$$

with the rotation matrix

$$\mathbf{R}_e^b(\phi, \theta, \psi) = \mathbf{R}_x(\phi) \mathbf{R}_y(\theta) \mathbf{R}_z(\psi) \quad (2.2)$$

which is the multiplication of rotation matrices about the individual axis x , y and z defined as

$$\mathbf{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \quad (2.3)$$

$$\mathbf{R}_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (2.4)$$

$$\mathbf{R}_z(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.5)$$

This definition of the rotation order is also called Cardan angles and commonly used in the aerospace field[10]. The inverse rotation from the body fixed frame to the earth fixed frame is

$$\mathbf{R}_b^e = \mathbf{R}_e^{b^{-1}} = \mathbf{R}_e^{b^T}. \quad (2.6)$$

Due to the rotations one after the other about the z , y and then about the x axis, the derivative of the Euler angles with respect to the angular velocity in the body frame derived by [10] is

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix}}_{:=S} \boldsymbol{\omega}_b. \quad (2.7)$$

It is visible, that at angles of $\pm 90^\circ$ the denominator of the elements of the third row become zero and the matrix becomes singular. This means no unique solution to calculate the angular velocity in the earth frame is available. A possible solution of this problem is to define the frame, so that no gimbal lock occurs, respectively no angle of $\pm 90^\circ$ for the angle θ occurs.

Unit quaternions

A second representation of a rotation is the unit quaternion. This is a quaternion with a special constraint. First the quaternion operations are presented and then the constraint to create a unit quaternion. A quaternion is presented by a four dimensional vector

$$\mathbf{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}. \quad (2.8)$$

This vector can be splitted up into a scalar part

$$q_s = q_0 \quad (2.9)$$

and a 3 dimensional vector part

$$\mathbf{q}_v = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}. \quad (2.10)$$

Due to the four dimensional representation, no singularities can occur[10]. Another benefit of the quaternion representation is, the calculations of them is more accurate and faster than using Euler angles because of the accuracy and the calculation time of trigonometric functions. No trigonometric functions are used in the quaternion representation[10]. There are several different definitions of the multiplication of quaternions [11, 12, 13] which are all equivalent. In this thesis the definition of J. M. Maley

$$\mathbf{p} \otimes \mathbf{q} = \begin{bmatrix} p_s q_s - \mathbf{p}_v \cdot \mathbf{q}_v \\ p_s \mathbf{q}_v + q_s \mathbf{p}_v + \mathbf{p}_v \times \mathbf{q}_v \end{bmatrix} \quad (2.11)$$

is used [11]. The norm of a quaternion is defined same as the norm for normal vectors as

$$\|\mathbf{q}\| = \sqrt{q_s^2 + q_{v1}^2 + q_{v2}^2 + q_{v3}^2} \quad (2.12)$$

and the adjoint operation is defined as

$$\bar{\mathbf{q}} = \begin{bmatrix} q_s \\ -\mathbf{q}_v \end{bmatrix}. \quad (2.13)$$

This function is used in the inverse operation which is defined as the adjoint of the quaternion divided by the norm of a quaternion

$$\mathbf{q}^{-1} = \frac{\bar{\mathbf{q}}}{\|\mathbf{q}\|}. \quad (2.14)$$

A unit quaternion is a quaternion with the constraint that the norm of the quaternion is equal to one. With this constraint, the multiplication of unit quaternions $\mathbf{p} \otimes \mathbf{p}$ can be illustrated as a rotation of the unit quaternion \mathbf{p} about the angle of \mathbf{q} . The result is again a unit quaternion. A three dimensional vector $\boldsymbol{\omega} \in \mathbb{R}^3$ can be seen as a quaternion with scalar part zero and as vector part the three dimensional vector[13]

$$\mathbf{q}_\omega = \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} \quad (2.15)$$

A scalar $s \in \mathbb{R}$ can be seen as a quaternion with scalar part s and vectorpart zero

$$\mathbf{q}_s = \begin{bmatrix} s \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (2.16)$$

Equivalent to the frame representation in the Euler angle representation, a vector $\boldsymbol{\omega}_e \in \mathbb{R}^3$ in the earth frame can be represented in the body frame by multiplying the vector from the left side with the unit quaternion which represents the rotation from the earth frame to the body frame and from the right side with its inverse

$$\begin{bmatrix} 0 \\ \boldsymbol{\omega}_b \end{bmatrix} = \mathbf{q} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega}_e \end{bmatrix} \otimes \mathbf{q}^{-1}. \quad (2.17)$$

This equation can be rewritten to reduce the number of calculations by omitting the scalar part. The result is

$$\boldsymbol{\omega}_b = \mathbf{R}_e^b(\mathbf{q})\boldsymbol{\omega}_e \quad (2.18)$$

with the quaternion rotation matrix

$$\mathbf{R}_e^b(\mathbf{q}) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2q_1q_2 + 2q_0q_3 & 2q_1q_3 - 2q_0q_2 \\ 2q_1q_2 - 2q_0q_3 & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2q_2q_3 + 2q_0q_1 \\ 2q_1q_3 + 2q_0q_2 & 2q_2q_3 - 2q_0q_1 & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}. \quad (2.19)$$

To rotate a vector from the body frame to the earth frame, the matrix (2.19) must be inverted, which is equivalent to the transpose of the matrix

$$\boldsymbol{\omega}_e = \mathbf{R}_e^b(\mathbf{q})^T \boldsymbol{\omega}_b. \quad (2.20)$$

To calculate the Euler angles from a unit quaternion

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \text{atan2}(2q_2q_3 + 2q_0q_1, q_3^2 - q_2^2 - q_1^2 + q_0^2) \\ -\text{asin}(2q_1q_3 - 2q_0q_2) \\ \text{atan2}(2q_1q_2 + 2q_0q_3, q_1^2 + q_0^2 - q_3^2 - q_2^2) \end{bmatrix} \quad (2.21)$$

is used. An important unit quaternion is the identity quaternion

$$\mathbf{q}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.22)$$

which is represented with the scalar part equal to one and the vector part equal to zero. Using (2.21) it can be seen, that the identity quaternion corresponds to the attitude with all angles ϕ, θ, ψ zero

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \quad (2.23)$$

The reverse calculation from Euler angles to quaternions is defined in [10] as

$$\mathbf{q} = \begin{bmatrix} \cos(\phi/2)\cos(\theta/2)\cos(\psi/2) + \sin(\phi/2)\sin(\theta/2)\sin(\psi/2) \\ -\cos(\phi/2)\sin(\theta/2)\sin(\psi/2) + \cos(\theta/2)\cos(\psi/2)\sin(\phi/2) \\ \cos(\phi/2)\cos(\psi/2)\sin(\theta/2) + \sin(\phi/2)\cos(\theta/2)\sin(\psi/2) \\ \cos(\phi/2)\cos(\theta/2)\sin(\psi/2) - \sin(\phi/2)\cos(\psi/2)\sin(\theta/2) \end{bmatrix}. \quad (2.24)$$

The time derivative of a unit quaternion can be expressed in terms of the unit quaternion itself and the angular velocity in the body frame as

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega}_b \end{bmatrix} \quad (2.25)$$

Integrating the discrete angular velocity to retrieve the attitude in the unit quaternion representation can be done by using

$$\mathbf{q}_k = e^{\frac{1}{2}\bar{\boldsymbol{\omega}}_x \Delta t} \mathbf{q}_{k-1} \quad (2.26)$$

with $\bar{\boldsymbol{\omega}}_x$ the cross product matrix

$$\boldsymbol{\omega}_x = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \quad (2.27)$$

of the mean of the angular velocity defined as

$$\bar{\boldsymbol{\omega}} = \frac{\boldsymbol{\omega}_{b_k} + \boldsymbol{\omega}_{b_{k-1}}}{2} \quad (2.28)$$

where $\boldsymbol{\omega}_{b_k}$ defines the angular velocity at step k and $\boldsymbol{\omega}_{b_{k-1}}$ the angular velocity at step $k-1$.

2.2 Sensors

The AWES, which is developed, has multiple sensors on board to estimate the attitude, the position and the velocity of the kite. In this chapter the used sensors are explained.

2.2.1 Accelerometer

The accelerometer measures the acceleration of the body. In steady state with no movement, the accelerometer measures the gravity

$$\mathbf{g}_e = \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} \quad (2.29)$$

[m s⁻²], but while moving and rotating the sensor measures two additional accelerations, the translational $\dot{\mathbf{v}}_b$ and the centripetal acceleration \mathbf{a}_{cp} [8]. The acceleration in body coordinates is defined as

$$\mathbf{a}_b = \dot{\mathbf{v}}_b + \mathbf{a}_{cp} - \mathbf{g}_b \quad (2.30)$$

with the centripetal acceleration

$$\mathbf{a}_{cp} = \boldsymbol{\omega}_b \times \mathbf{v}_b \quad (2.31)$$

and the gravity vector in body coordinates \mathbf{g}_b . \mathbf{v}_b describes the velocity in the body coordinates. The accelerometer is not able to determine all three Euler angles itself, only the roll and pitch angle are determinable. Additional sensors are required to determine the complete attitude. The measured acceleration is the real acceleration plus an additive noise

$$\mathbf{a}_{b_{meas}} = \mathbf{a}_b + \boldsymbol{\eta}_{a_b}. \quad (2.32)$$

2.2.2 Gyroscope

A gyroscope is a sensor, many times combined in one module with the accelerometer, which measures the angular rate about a specific axis. This means, the measurements are in the body fixed frame. An Inertial Measurement Unit (IMU) based sensor combines three orthogonal gyroscopes together to get the angular rates for the roll, pitch and yaw axis. The calculation of the Euler angles / quaternions from angular rates are already described in (2.7) and (2.25). Due to gyroscope biases the integration of this values lead to drift in the attitude, whereas absolute angle measurement units like the magnetometer in 2.2.3 and the accelerometer in 2.2.1 are necessary. The gyroscope is used because of its high sample rate to integrate the angles, when any measurements from the accelerometer or magnetometer are available. The gyroscope measurements with gyroscope bias and noise can be described as

$$\omega_{b_{meas}} = \omega_b + b_w + \eta_{\omega_b}. \quad (2.33)$$

with b_ω the gyroscope bias and η_{ω_b} the additive white gaussian noise

2.2.3 Magnetometer

One possibility to determine the yaw angle, is a magnetometer, which measures the strength and the direction of the earth magnetic field. If the x component of the magnetometer directs in the direction of the magnetic field, the magnetometer would measure a magnetic field only on the x-Axis

$$\mathbf{m}_{mag} = \begin{bmatrix} B \\ 0 \\ 0 \end{bmatrix} \quad (2.34)$$

with B the strength of the field. The magnetic north is not aligned with the geographic north[14, 15] and therefore, to receive the magnetic field in the NED frame the magnetic vector must be rotated by the inclination angle δ_i about the y-Axis and by the declination angle δ_d about the z-Axis

$$\mathbf{m}_e = \mathbf{R}_{mag}^e \mathbf{m}_{mag} = \mathbf{R}_z(\delta_d) \mathbf{R}_y(\delta_i) \mathbf{m}_{mag} \quad (2.35)$$

with the rotation matrices defined in 2.1.3. The inclination angle vary over the earth surface, at south with -90° to $+90^\circ$ in the north which is illustrated in figure 2.3. The angles can be determined by positioning the magnetometer in the direction of geographic north on flat ground with pitch and roll angle equal to zero and using the equations

$$\delta_i = \frac{180 \arcsin(m_{b_z})}{\pi} \quad (2.36)$$

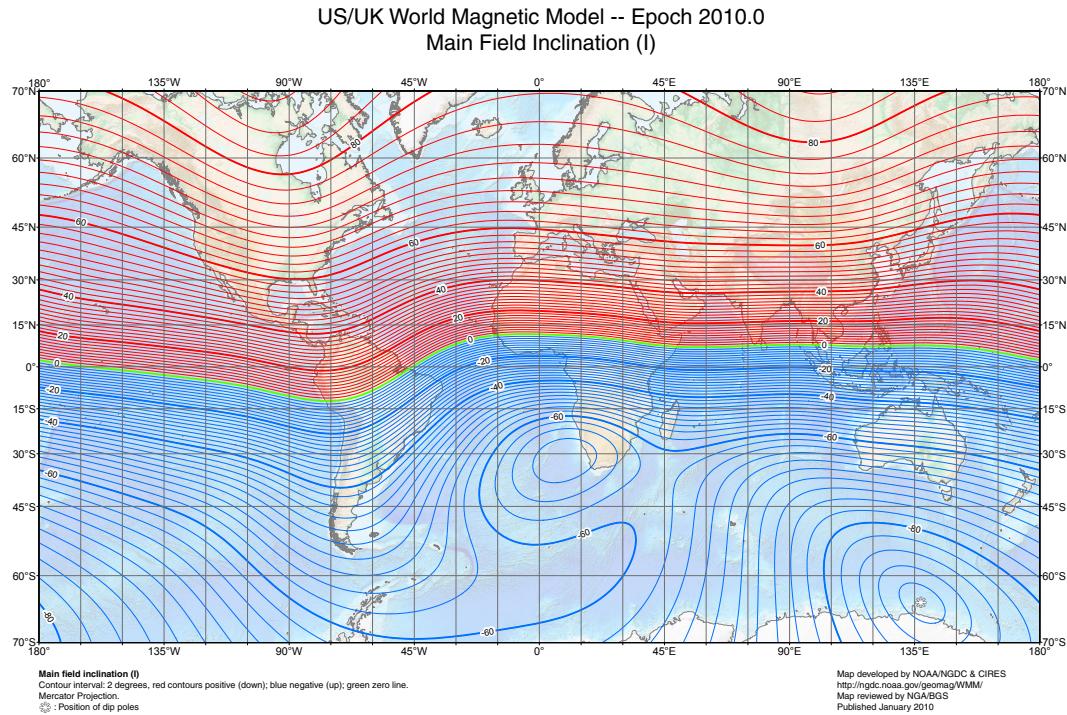
and

$$\delta_d = \frac{180 \arccos\left(\frac{m_{b_x}}{\sqrt{1-m_{b_z}^2}}\right)}{\pi} \quad (2.37)$$

or by using world magnetic models [15, 16].

Calibration

Due to the environment or magnetic distortion through electronic devices the measured direction is not aligned with the magnetic field, which means, the magnetometer must be calibrated to compensate these effects. The magnetic distortion can be classified into two main distortions, the hard-iron and the soft-iron magnetic distortion[18].



Source: [17]

Figure 2.3: Magnetic inclination map.

Hard-iron Calibration

The hard-iron distortion comes mainly from ferromagnetic components on the Printed Circuit Board (PCB) or other ferromagnetic components like cases or electric motors next to the sensor [19, 20]. These components rotate together with the sensor and therefore this distortion is an additive offset

$$\mathbf{m}_{b_{meas}} = \mathbf{m}_b + \mathbf{m}_{b_{HI}} \quad (2.38)$$

with $\mathbf{m}_{b_{HI}}$ the hard iron distortion. The offsets and the strength of the magnetic field can be determined by using the previous measured data and the method from Ozyagcilar [20]. A least square minimization of the residual

$$\mathbf{r} = \mathbf{Y} - \mathbf{X}\boldsymbol{\beta} \quad (2.39)$$

with \mathbf{Y} , \mathbf{X} and $\boldsymbol{\beta}$ defined as

$$\mathbf{y} = \begin{bmatrix} m_{b_{meas_x}}[0]^2 + m_{b_{meas_y}}[0]^2 + m_{b_{meas_z}}[0]^2 \\ m_{b_{meas_x}}[1]^2 + m_{b_{meas_y}}[1]^2 + m_{b_{meas_z}}[1]^2 \\ \vdots \\ m_{b_{meas_x}}[M-1]^2 + m_{b_{meas_y}}[M-1]^2 + m_{b_{meas_z}}[M-1]^2 \end{bmatrix} \quad (2.40)$$

$$\mathbf{X} = \begin{bmatrix} m_{b_{meas_x}}[0] & m_{b_{meas_y}}[0] & m_{b_{meas_z}}[0] & 1 \\ m_{b_{meas_x}}[1] & m_{b_{meas_y}}[1] & m_{b_{meas_z}}[1] & 1 \\ \vdots & \vdots & \vdots & \vdots \\ m_{b_{meas_x}}[M-1] & m_{b_{meas_y}}[M-1] & m_{b_{meas_z}}[M-1] & 1 \end{bmatrix} \quad (2.41)$$

$$\boldsymbol{\beta} = \begin{bmatrix} 2B_{HI_x} \\ 2B_{HI_y} \\ 2B_{HI_z} \\ B^2 - B_{HI_x}^2 - B_{HI_y}^2 - B_{HI_z}^2 \end{bmatrix} \quad (2.42)$$

is made. The solution of this minimization problem is

$$\boldsymbol{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}. \quad (2.43)$$

After solving this equation, the hard-iron offsets can be calculated with equation (2.42).

Soft-iron Calibration

The soft iron distortion is not anymore a simple offset. The soft iron distortion depends on the environment which is not rotating with the sensor. So this calibration must be done every time the environment around the sensor changed. The task of this calibration is to find the soft-iron distortion matrix \mathbf{W} which is defined as

$$\mathbf{m}_{b_{meas}} = \mathbf{W} \mathbf{m}_b + \mathbf{m}_{b_{HI}}. \quad (2.44)$$

The strength of the earth magnetic field does not change when rotating the magnetometer. Therefore, the magnetic field $\mathbf{m}_b = \mathbf{W}^{-1}(\mathbf{m}_{b_{meas}} - \mathbf{m}_{b_{HI}})$ must fit

$$(\mathbf{W}^{-1}(\mathbf{m}_{b_{meas}} - \mathbf{m}_{b_{HI}}))^T \mathbf{W}^{-1}(\mathbf{m}_{b_{meas}} - \mathbf{m}_{b_{HI}}) = B^2 \quad (2.45)$$

which describes an ellipsoid

$$(\mathbf{r} - \mathbf{r}_0)^T \mathbf{A}(\mathbf{r} - \mathbf{r}_0) = const \quad (2.46)$$

with $\mathbf{A} = (\mathbf{W}^{-1})^T \mathbf{W}^{-1}$, $\mathbf{R} = \mathbf{m}_{b_{meas}}$ and center $\mathbf{r}_0 = \mathbf{m}_{b_{HI}}$ [19]. Through ellipsoid fitting the matrix \mathbf{A} can be determined[21, 22, 23]. If there aren't any other constraints to the matrix \mathbf{W}^{-1} it is not possible to calculate \mathbf{W}^{-1} uniquely, because the description of a sphere is independent of the rotation. To solve this problem T. Ozyagclar [19] used the constraint, that the matrix \mathbf{W}^{-1} is symmetric and so the soft iron matrix calculates to

$$\mathbf{W}^{-1} = \mathbf{A}^{\frac{1}{2}}. \quad (2.47)$$

Scaling the ellipsoid without using the symmetric constraint can be done by rotating the ellipsoid, to match the magnetometer axis, scaling the ellipsoid by its radii and then rotating back to its original rotation[24]. This can be done easily, because the eigenvectors of the ellipsoid are the axis of the ellipsoid. The soft iron matrix is then calculated as

$$\mathbf{W}^{-1} = \mathbf{R}_{ev} \mathbf{S} \mathbf{R}_{ev}^T \quad (2.48)$$

with

$$\mathbf{R}_{ev} = [\mathbf{e}\mathbf{v}_1 \quad \mathbf{e}\mathbf{v}_2 \quad \mathbf{e}\mathbf{v}_3] \quad (2.49)$$

the rotation matrix with the eigenvectors as entries and

$$\mathbf{S} = \begin{bmatrix} r_1 & 0 & 0 \\ 0 & r_2 & 0 \\ 0 & 0 & r_3 \end{bmatrix} \quad (2.50)$$

with r_x , $x \in \{1, 2, 3\}$ the radii of the ellipsoid, which are the inverse of the square roots of the eigenvalues

$$r_x = \frac{1}{\sqrt{|\lambda_x|}} \quad (2.51)$$

2.2.4 GPS

GPS is a global position navigation system developed by the United States using multiple satellites to determine the position and the velocity of the GPS receiver. The more GPS satellites are in line of sight, the more accurate are the measurements, but at least four are needed[25]. The measured values are in the inertia frame and must be transformed into the NED-frame by subtracting the position of the base in inertia coordinates

$$\mathbf{p}_{GPS} = \mathbf{p}_{GPS_{meas}} - \mathbf{p}_{base}. \quad (2.52)$$

The GPS position and velocity are signals with additive Gaussian noise

$$\begin{bmatrix} \mathbf{p}_{GPS_{meas}} \\ \mathbf{v}_{GPS_{meas}} \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{GPS} \\ \mathbf{v}_{GPS} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\eta}_{p_{GPS}} \\ \boldsymbol{\eta}_{v_{GPS}} \end{bmatrix}. \quad (2.53)$$

2.2.5 Barometer

The barometer is a sensor to measure the air pressure. This sensor must be placed in a place, where no wind blows to not falsify the measured data. It is used to calculate the height with the international barometric formula

$$h = 44330 \left(1 - \left(\frac{p}{p_0} \right)^{\frac{1}{5.255}} \right) \quad (2.54)$$

from [26] with p_0 the pressure at sea level and p the measured pressure. The pressure is a signal with additive Gaussian white noise

$$p_{meas} = p + \eta_p. \quad (2.55)$$

The height measured with the barometer is much more accurate and has a higher sample rate than the height measured with the GPS module, but over a longer time the barometer measured height can drift due to variations in the local barometric pressure[8]. To solve this problem, the barometer can be recalibrated with the height from the GPS module.

2.2.6 Tether angle sensor

The tether angle sensor consists of two potentiometers, which measure the azimuth and the elevation angle of the tether to the NED-frame. With the assumption, the tether is a straight line and knowing the tether length, the position of the kite can be calculated as

$$\mathbf{p}_{tether} = r \begin{bmatrix} \sin(\theta_{base})\sin(\phi_{base}) \\ \sin(\theta_{base})\cos(\phi_{base}) \\ \cos(\theta)_{base} \end{bmatrix} \quad (2.56)$$

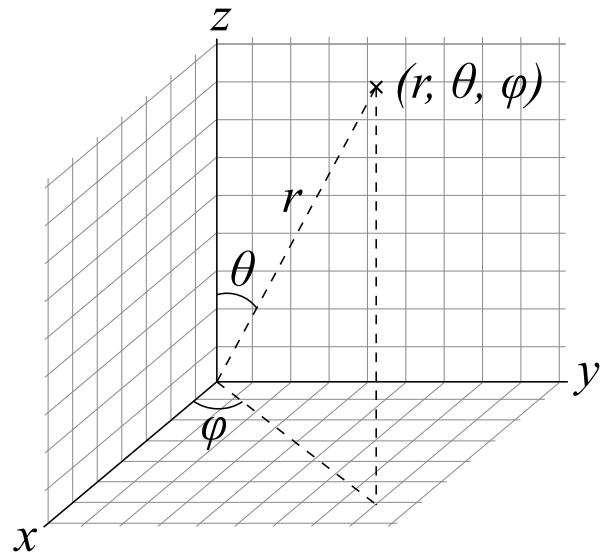
with θ_{base} the polar angle, ϕ_{base} the azimuth angle and r the tether length. Due to the mass of the tether the assumption of a straight line is not correct. This and a more correct model is discussed in section 3.3.3. The measured angles and tether length are distorted by additive Gaussian noise

$$r_{meas} = r + \eta_r \quad (2.57)$$

$$\theta_{base_{meas}} = \theta_{base} + \eta_{\theta_{base}} \quad (2.58)$$

$$\phi_{base_{meas}} = \phi_{base} + \eta_{\phi_{base}} \quad (2.59)$$

A second tether angle sensor is mounted on the kite and measures the angle of the tether coming into the kite. One reason of this sensor is described in section 3.3.3 where a model for tether sagging is proposed.



Source: [27]

Figure 2.4: Calculation of the position of the kite with the tether angle sensor.

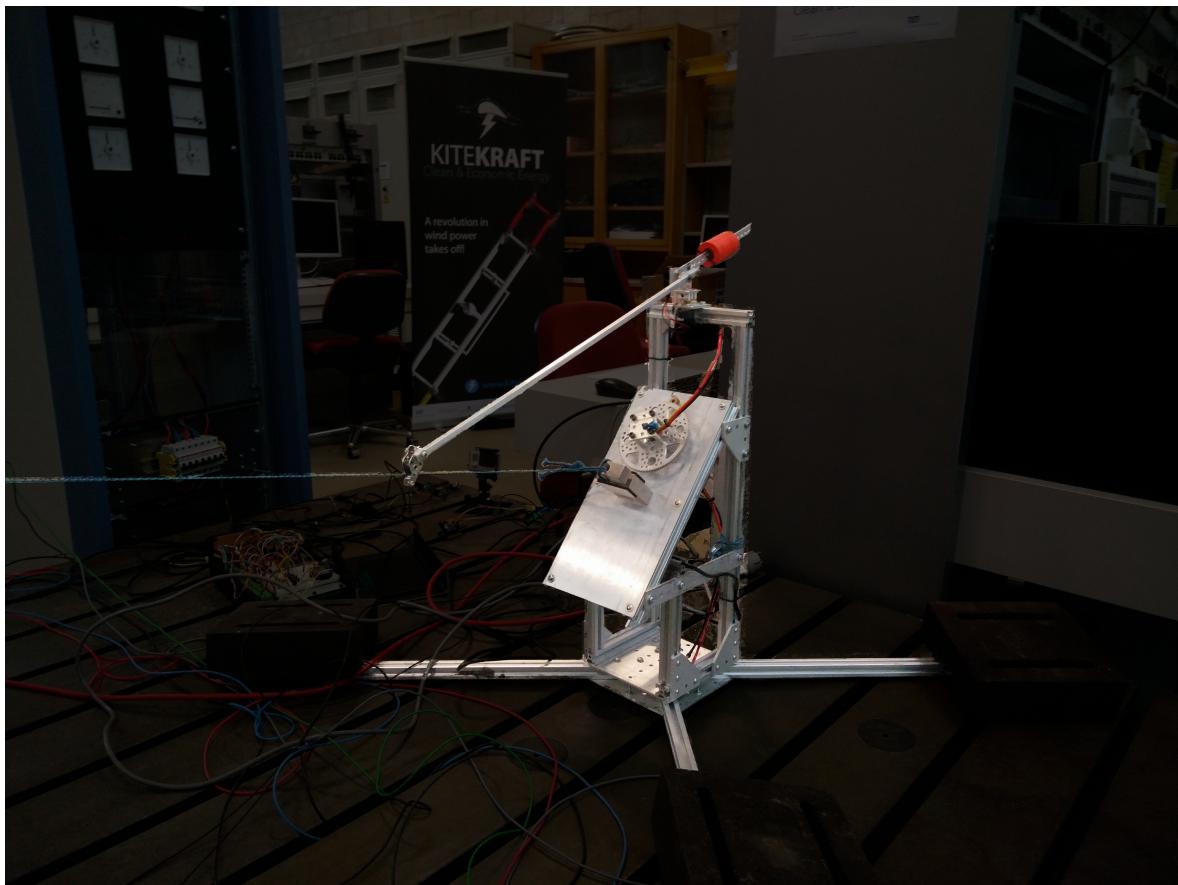


Figure 2.5: Tether angle sensor with tether.

2.3 Measurement systems

2.3.1 Attitude and heading reference system (AHRS)

An AHRS is a measurement system, which use magnetometer, accelerometer and gyroscope, called IMU, to estimate the attitude of the kite with respect to an earth fixed frame. The AHRS uses the position and the velocity only to compensate additional accelerations like the centripetal acceleration presented in 2.2.1, but the position and the velocity itself are derived from the GPS module, respectively the height from the barometer without fusing them together[8]. In figure 2.6 a blockdiagram of the AHRS is illustrated.

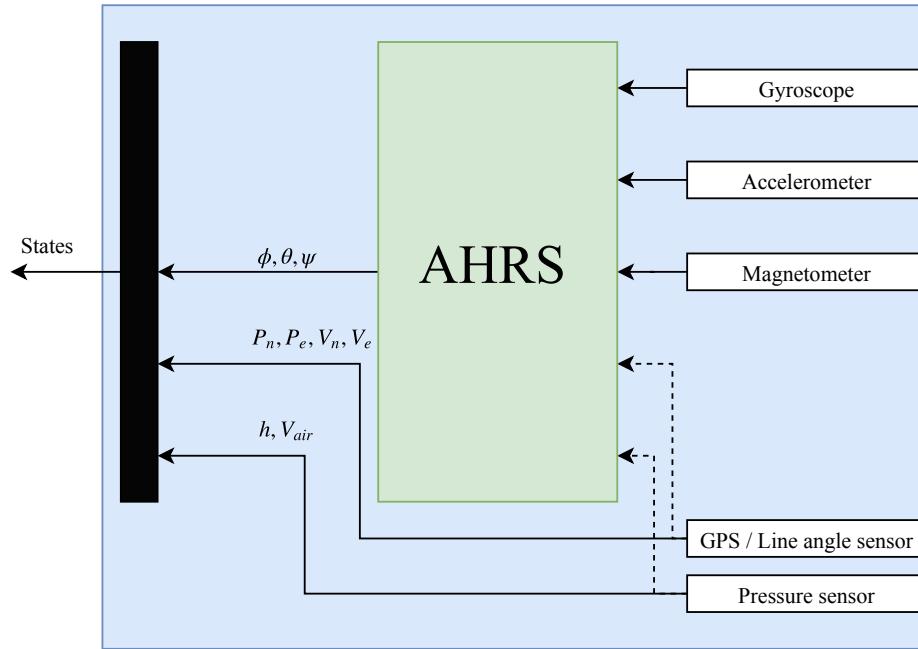


Figure 2.6: Block diagram of an AHRS, GPS and a Barometer to estimate all degrees of freedom.

2.3.2 Inertial navigation system (INS)

An INS uses position, velocity sensors as well as an IMU to estimate the attitude, position and velocity in one filter. This increases the accuracy of the estimation, because in the above type of filter, any translational acceleration is compensated and so for higher translational acceleration the estimation accuracy decreases. The drawback of this filter is the higher complexity and so higher computational performance is needed.

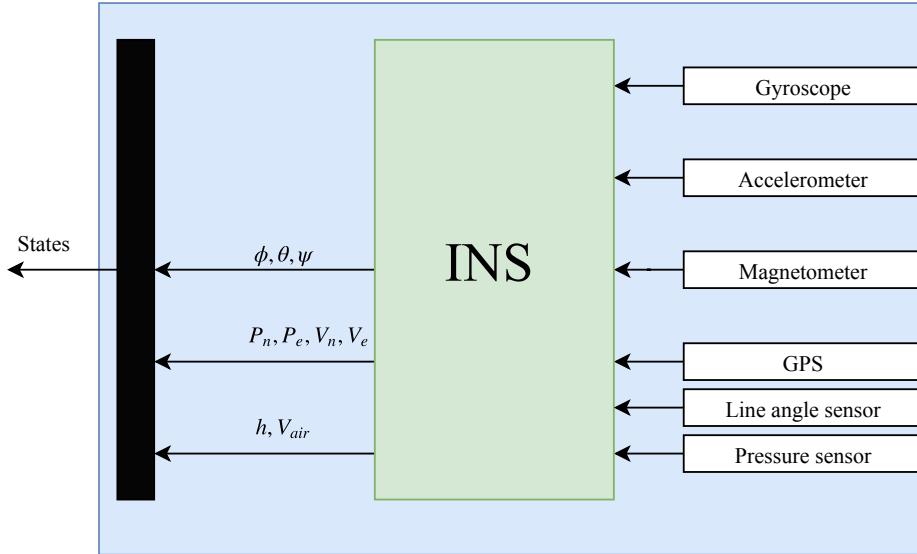


Figure 2.7: State estimation with a full GPSINS.

2.4 Filter

In this section different filter types are presented which were used in this thesis. The first one is the Kalman filter and with the extension for nonlinear systems. Second is the Mahony complementary filter[28] which in contrast to the EKF is a real nonlinear filter.

2.4.1 Kalman filter

In this section a brief derivation of the Kalman filter is presented, because it is one of the main filters used in this thesis. After the derivation a comparsion between the time-discrete Kalman filter and the discretized Kalman-Bucy filter is made.

Linear Kalman Filter

The (linear) Kalman filter is a recursive filter which is based on estimating unknowns states of a linear system

$$\mathbf{X}_n = \mathbf{G}_n \mathbf{X}_{n-1} + \mathbf{B}_n \mathbf{u}_n + \mathbf{V}_n \quad (2.60)$$

$$\mathbf{Y}_n = \mathbf{H}_n \mathbf{X}_n + \mathbf{W}_n \quad (2.61)$$

with \mathbf{V}_n the process noise and \mathbf{W}_n the measurement noise[29, 30] by minimizing the mean squared error (MSE) with a prior information about the noise.

In general the mean of a random variable X is defined as

$$E[X] = \int_{\mathbb{X}} x f_X(x) dx \quad (2.62)$$

with $f_X(x)$ the PDF of the random variable X . The variance of a random variable X is defined as

$$VAR[X] = E[(X - E[X])^2] \quad (2.63)$$

which can be extended for a multivariate random variable \mathbf{X} to

$$C_{\mathbf{X}} = C[\mathbf{X}, \mathbf{X}] = E [(\mathbf{X} - E[\mathbf{X}])(\mathbf{X} - E[\mathbf{X}])^T]. \quad (2.64)$$

The MSE of an estimator $T(X)$ is defined as

$$\epsilon[T(X)] = E \left[(T(X) - \Theta)^2 \right] \quad (2.65)$$

with Θ the real value, which should be estimated. The PDF of the real value is defined as

$$f_{\Theta}(\theta; \sigma) \quad (2.66)$$

with θ the mean and σ the variance of the random variable Θ . The PDF of the random variable X with the knowledge of Θ is defined as

$$f_{X|\Theta}(x|\theta). \quad (2.67)$$

Minimizing the MSE by using two times the definition of the mean (2.62)

$$\begin{aligned} \epsilon[T(X)] &= \int_{\Theta} \int_{\mathbb{X}} (T(x) - \theta)^2 f_{X|\Theta}(x|\theta) dx f_{\Theta}(\theta; \sigma) d\theta \\ &= \underbrace{\int_{\mathbb{X}} \int_{\Theta} (T(x) - \theta)^2 f_{\Theta|X}(\theta|x) d\theta}_{\text{minimize}} \underbrace{f_X(x)}_{\text{independent of } \Theta} dx \\ &\Rightarrow \min_{T(x)} \int_{\Theta} (T(x) - \theta)^2 f_{\Theta|X}(\theta|x) d\theta \end{aligned} \quad (2.68)$$

results in an estimator called conditional mean estimator or Bayes estimator[29, 31]

$$T(x) = \int_{\Theta} \theta f_{\Theta|X}(\theta|x) d\theta = E[\Theta|X = x]. \quad (2.69)$$

This estimator minimizes the MSE. The Kalman filter estimates random variables in a recursive manner using this estimator.

The Kalman filter is usable at the following assumptions

Assumption 1. \mathbf{X}_m and \mathbf{V}_n are stochastically independent for any m, n :

$$E[\mathbf{X}_m \mathbf{V}_n] = E[\mathbf{X}_m] E[\mathbf{V}_n]$$

Assumption 2. The process noise \mathbf{V}_n and the measurement noise \mathbf{W}_n are gaussian white noises with zero mean

$$\begin{aligned} \mathbf{V}_n &\sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\mathbf{V}_n}) \\ \mathbf{W}_n &\sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\mathbf{W}_n}) \end{aligned}$$

Assumption 3. The random variables $\mathbf{V}_k, \mathbf{V}_l$ and $\mathbf{W}_m, \mathbf{W}_n$ are stochastically independent for any k, l and m, n :

$$\begin{aligned} E[\mathbf{V}_k \mathbf{V}_l] &= E[\mathbf{V}_k] E[\mathbf{V}_l] \\ E[\mathbf{W}_m \mathbf{W}_n] &= E[\mathbf{W}_m] E[\mathbf{W}_n] \end{aligned}$$

Assumption 4. The noises \mathbf{V}_n and \mathbf{W}_m are stochastically independent for any n and m :

$$E[\mathbf{V}_n \mathbf{W}_m] = E[\mathbf{V}_n] E[\mathbf{W}_m]$$

The Kalman filter can be divided into two steps, the prediction step and the corrector step[29]. In the prediction step the expectation of the state \mathbf{X}_n with the knowledge of $\mathbf{Y}_{(n-1)}$, which defines all observations until the step n-1, is defined as

$$\hat{\mathbf{x}}_{n|n-1} = E[\mathbf{X}_n | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)}]. \quad (2.70)$$

The covariance of \mathbf{X}_n is defined in [29] as

$$C_{\mathbf{X}_{n|n-1}} = E \left[(\mathbf{X}_n - \hat{\mathbf{x}}_{n|n-1}) (\mathbf{X}_n - \hat{\mathbf{x}}_{n|n-1})^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} \right]. \quad (2.71)$$

$\hat{\mathbf{x}}_{n|n-1}$ can be derived from the Chapman-Kolmogorov equation [29] as

$$\begin{aligned}
 \hat{\mathbf{x}}_{n|n-1} &= \underbrace{\int_{\mathbb{X}} \mathbf{x}_n f_{\mathbf{X}_n|\mathbf{Y}_{(n-1)}}(\mathbf{x}_n|\mathbf{y}_{(n-1)}) d\mathbf{x}_n}_{\text{estimator to minimize the MSE (2.68)}} \\
 &= \int_{\mathbb{X}} \mathbf{x}_n \int_{\mathbb{X}} f_{\mathbf{X}_n|\mathbf{X}_{n-1}}(\mathbf{x}_n|\mathbf{x}_{n-1}) \underbrace{f_{\mathbf{X}_{n-1}|\mathbf{Y}_{(n-1)}}(\mathbf{x}_{n-1}|\mathbf{y}_{(n-1)})}_{\text{independent from } \mathbf{x}_n} d\mathbf{x}_{n-1} d\mathbf{x}_n \\
 &= \int_{\mathbb{X}} \int_{\mathbb{X}} \underbrace{\mathbf{x}_n f_{\mathbf{X}_n|\mathbf{X}_{n-1}}(\mathbf{x}_n|\mathbf{x}_{n-1}) d\mathbf{x}_n}_{E[\mathbf{x}_n|\mathbf{x}_{n-1}] = \mathbf{G}_n \mathbf{x}_{n-1} + \mathbf{B}_n \mathbf{u}_n (2.60)} f_{\mathbf{X}_{n-1}|\mathbf{Y}_{(n-1)}}(\mathbf{x}_{n-1}|\mathbf{y}_{(n-1)}) d\mathbf{x}_{n-1} \\
 &= \int_{\mathbb{X}} \mathbf{G}_n \mathbf{x}_{n-1} f_{\mathbf{X}_{n-1}|\mathbf{Y}_{(n-1)}}(\mathbf{x}_{n-1}|\mathbf{y}_{(n-1)}) d\mathbf{x}_{n-1} + \mathbf{B}_n \mathbf{u}_n \underbrace{\int_{\mathbb{X}} f_{\mathbf{X}_{n-1}|\mathbf{Y}_{(n-1)}}(\mathbf{x}_{n-1}|\mathbf{y}_{(n-1)}) d\mathbf{x}_{n-1}}_{=1} \\
 &= \mathbf{G}_n \hat{\mathbf{x}}_{n-1|n-1} + \mathbf{B}_n \mathbf{u}_n. \tag{2.72}
 \end{aligned}$$

The conditional state covariance matrix also called error covariance [32] of \mathbf{X}_n can be derived as

$$\begin{aligned}
 \mathbf{C}_{X_{n|n-1}} &= E \left[(\mathbf{X}_n - \hat{\mathbf{x}}_{n|n-1}) (\mathbf{X}_n - \hat{\mathbf{x}}_{n|n-1})^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} \right] \\
 &= E \left[(\mathbf{G}_n \mathbf{X}_{n-1} + \mathbf{B}_n \mathbf{u}_n + \mathbf{V}_n - (\mathbf{G}_n \hat{\mathbf{x}}_{n-1|n-1} + \mathbf{B}_n \mathbf{u}_n)) \right. \\
 &\quad \left. (\mathbf{G}_n \mathbf{X}_{n-1} + \mathbf{V}_n - (\mathbf{G}_n \hat{\mathbf{x}}_{n-1|n-1} + \mathbf{B}_n \mathbf{u}_n))^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} \right] \\
 &= E \left[\mathbf{G}_n (\mathbf{X}_n - \hat{\mathbf{x}}_{n-1|n-1}) (\mathbf{X}_n - \hat{\mathbf{x}}_{n-1|n-1})^T \mathbf{G}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} v \right] \\
 &\quad - \underbrace{E \left[\mathbf{V}_n \hat{\mathbf{x}}_{n-1|n-1}^T \mathbf{G}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} \right]}_{= E[\mathbf{V}_n] \hat{\mathbf{x}}_{n-1|n-1}^T \mathbf{G}_n^T = 0} - \underbrace{E \left[\mathbf{G}_n \hat{\mathbf{x}}_{n-1|n-1} \mathbf{V}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} \right]}_{= \mathbf{G}_n \hat{\mathbf{x}}_{n-1|n-1} E[\mathbf{V}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)}] = 0} \\
 &\quad + \underbrace{E \left[\mathbf{G}_n \mathbf{X}_n \mathbf{V}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} \right]}_{= \mathbf{G}_n E[\mathbf{X}_n] E[\mathbf{V}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)}] = 0} + \underbrace{E \left[\mathbf{V}_n \mathbf{X}_n^T \mathbf{G}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} \right]}_{= \mathbf{V}_n E[\mathbf{X}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)}] \mathbf{G}_n^T = 0} \tag{2.73} \\
 &\quad + E \left[\mathbf{V}_n \mathbf{V}_n^T | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)} \right] \\
 &= \mathbf{G}_n \mathbf{C}_{X_{n-1|n-1}} \mathbf{G}_n^T + \mathbf{C}_{\mathbf{V}_n}. \tag{2.74}
 \end{aligned}$$

In the corrector step the Mean of \mathbf{X}_n respect to $\mathbf{Y}_{(n)}$ will be calculated with the Bayes rule[29]

$$f_{\mathbf{X}_n|\mathbf{Y}_{(n)}}(\mathbf{x}_n|\mathbf{y}_n) = \frac{f_{\mathbf{Y}_n|\mathbf{X}_n}(\mathbf{y}_n|\mathbf{x}_n) f_{\mathbf{X}_n|\mathbf{Y}_{(n-1)}}(\mathbf{x}_n|\mathbf{y}_{n-1})}{f_{\mathbf{Y}_n|\mathbf{Y}_{(n-1)}}(\mathbf{y}_n|\mathbf{y}_{n-1})}. \tag{2.75}$$

With the assumption 2 all probability density functions of (2.75) are gaussian. By comparing the exponents of (2.75) we get the equation

$$\begin{aligned}
 -\frac{1}{2} (\mathbf{x}_n - \hat{\mathbf{x}}_{n|n})^T \mathbf{C}_{\mathbf{X}_{n|n}}^{-1} (\mathbf{x}_n - \hat{\mathbf{x}}_{n|n}) &\propto -\frac{1}{2} (\mathbf{y}_n - \mu_{\mathbf{Y}_n|\mathbf{X}_n})^T \mathbf{C}_{\mathbf{Y}_n|\mathbf{X}_n}^{-1} (\mathbf{y}_n - \mu_{\mathbf{Y}_n|\mathbf{X}_n}) \\
 &\quad - \frac{1}{2} (\mathbf{x}_n - \hat{\mathbf{x}}_{n|n-1})^T \mathbf{C}_{\mathbf{X}_{n|n-1}}^{-1} (\mathbf{x}_n - \hat{\mathbf{x}}_{n|n-1}) \\
 &\Rightarrow \mathbf{x}_n^T \mathbf{C}_{\mathbf{X}_{n|n}}^{-1} \mathbf{x}_n - 2 \mathbf{x}_n^T \mathbf{C}_{\mathbf{X}_{n|n}}^{-1} \hat{\mathbf{x}}_{n|n} + \dots \propto \\
 &\underbrace{\mathbf{x}_n^T \left(\mathbf{C}_{\mathbf{X}_{n|n-1}}^{-1} + \mathbf{H}_n^T \mathbf{C}_{\mathbf{Y}_n|\mathbf{X}_n=\mathbf{x}_n}^{-1} \mathbf{H}_n \right) \mathbf{x}_n}_{\mathbf{C}_{\mathbf{X}_{n|n}}^{-1}} - \underbrace{2 \mathbf{x}_n^T \left(\mathbf{H}_n^T \mathbf{C}_{\mathbf{Y}_n|\mathbf{X}_n=\mathbf{x}_n}^{-1} \mathbf{y}_n + \mathbf{C}_{\mathbf{X}_{n|n-1}}^{-1} \hat{\mathbf{x}}_{n|n-1} \right)}_{\mathbf{C}_{\mathbf{X}_{n|n}}^{-1} \hat{\mathbf{x}}_{n|n}} + \dots \tag{2.76}
 \end{aligned}$$

By comparing the terms of (2.76) we get the covariance matrix

$$\begin{aligned}
 C_{\mathbf{X}_{n|n}} &= \left(\mathbf{H}_n^T \mathbf{C}_{\mathbf{Y}_{n|\mathbf{X}_n=\mathbf{x}_n}}^{-1} \mathbf{y}_n + \mathbf{C}_{\mathbf{X}_{n|n-1}}^{-1} \hat{\mathbf{x}}_{n|n-1} \right)^{-1} \\
 &= \mathbf{C}_{\mathbf{X}_{n|n-1}} - \mathbf{C}_{\mathbf{X}_{n|n-1}} \mathbf{H}_n^T \underbrace{\left(\underbrace{\mathbf{C}_{\mathbf{Y}_{n|\mathbf{X}_n=\mathbf{x}_n}}^{-1} + \mathbf{H}_n \mathbf{C}_{\mathbf{X}_{n|n-1}} \mathbf{H}_n^T}_{\text{Kalman Gain } \mathbf{K}_n} \right)^{-1}}_{\mathbf{C}_{\mathbf{W}_n}} \mathbf{H}_n \mathbf{C}_{\mathbf{X}_{n|n-1}} \\
 &= \mathbf{C}_{\mathbf{X}_{n|n-1}} - \mathbf{K}_n \mathbf{H}_n \mathbf{C}_{\mathbf{X}_{n|n-1}}
 \end{aligned} \tag{2.77}$$

with \mathbf{K}_n the Kalman gain. Replacing of $\mathbf{C}_{\mathbf{X}_{n|n}}$ in $\hat{\mathbf{x}}_{n|n}$ (2.76) with (2.77) we get

$$\hat{\mathbf{x}}_{n|n} = \hat{\mathbf{x}}_{n|n-1} + \mathbf{K}_n (\mathbf{y}_n - \mathbf{H}_n \hat{\mathbf{x}}_{n|n-1}). \tag{2.78}$$

With the equations (2.72), (2.73), (2.77) and (2.78) we get a recursive filter which minimizes the MSE. There are different other names for the process noise covariance matrix \mathbf{C}_{V_n} , often called \mathbf{Q} matrix and the measurement noise covariance matrix \mathbf{C}_{W_n} called \mathbf{R} matrix.

Stationary accuracy To calculate the stationary accuracy it is possible to use equation (2.78) and (2.70) and set $x_{n|n}$ equal to $x_{n|n-1}$. As a result of this procedure we get the stationary accuracy as

$$x_{\text{error}} = (\mathbf{I}_x - \mathbf{G})^{-1} (\mathbf{B}\mathbf{u} + \mathbf{C}\mathbf{M}_L) - (\mathbf{I}_x - (\mathbf{G} - \mathbf{K}\mathbf{H}\mathbf{G}))^{-1} \mathbf{D}. \tag{2.79}$$

Converting time-continuous system to time-discrete system Systems are in general time-continuous, but sampling of measurements with sensors, discrete samples arise. Therefore, the time-continuous system must be transferred to a time-discrete representation. This can be done with the method developed by C. F. van Loan [32]. Assuming a system

$$\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{v} \tag{2.80}$$

and

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{w} \tag{2.81}$$

which can be transferred into a discrete representation as shown in (2.60) and (2.61) with

$$\tilde{\mathbf{A}} = \begin{bmatrix} -\mathbf{A} & \mathbf{W} \\ \mathbf{0} & \mathbf{A}^T \end{bmatrix} \Delta t \tag{2.82}$$

$$\mathbf{B} = \text{expm}(\tilde{\mathbf{A}}) = \begin{bmatrix} \cdots & \mathbf{G}_n^{-1} \mathbf{V}_n \\ 0 & \mathbf{G}_n^T \end{bmatrix} \tag{2.83}$$

$$\mathbf{H}_n = \mathbf{C} \tag{2.84}$$

$$\mathbf{W}_n = \frac{\text{VAR}[w]}{\Delta t}. \tag{2.85}$$

The transition matrix \mathbf{G}_n can be determined by transposing the lower right part of the matrix \mathbf{B} , and the process noise matrix \mathbf{V}_n from multiplying the upper right part of \mathbf{B} with the transition matrix.

Comparing discrete Kalman filter with Kalman-Bucy Filter The original implementation of the Kalman filter was made with the assumption, of a discrete system. In some cases a time-continuous model is present and so the Kalman filter was extended to time-continuous models. This extension is named Kalman-Bucy Filter [33]. In this section a comparison between the time-discrete and the time-continuous model is made for a DC-motor because it is much simpler than an attitude estimator which must be linearized due to the nonlinearity of the rotation matrices. The block diagram for the comparison are visible in 2.8, 2.9 and 2.10. The discrete Kalman filter is driven from a function call with a frequency of 1000 Hz. The system of the DC-motor is described as

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \dot{i}(t) \\ \dot{\omega}(t) \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{c_M \Psi_n}{L} \\ \frac{c_M \Psi_n}{J} & 0 \end{bmatrix} \begin{bmatrix} i(t) \\ \omega(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} \eta_i \\ \eta_{\omega} \end{bmatrix} M_L(t) \quad (2.86)$$

and

$$y(t) = [1 \ 0] \begin{bmatrix} i(t) \\ \omega(t) \end{bmatrix} + \eta_i = i(t) + \eta_i \quad (2.87)$$

with the wire resistance R , the inductance L , the magnetic flux Ψ_n and the motor constant c_M [34]. Gaussian noises η_i , η_{ω} and η_i are added to the ideal system. The values of these constants and the covariance matrix entries are used from F. Bauer[34] as

$$R = 0.3 \Omega \quad (2.88)$$

$$L = 3 \text{ mH} \quad (2.89)$$

$$c_M \Psi_n = 1.13 \text{ V s} \quad (2.90)$$

$$J = 0.2 \text{ kg m}^2 \quad (2.91)$$

$$\mathbf{C}_V = \begin{bmatrix} VAR[\eta_i] & 0 \\ 0 & VAR[\eta_{\omega}] \end{bmatrix} = \begin{bmatrix} 10000 & 0 \\ 0 & 100 \end{bmatrix} \quad (2.92)$$

$$C_W = VAR[\eta_i] = 10 \quad (2.93)$$

The result can be seen in figure 2.11 and 2.12. There is a difference between the time-continuous Kalman-Bucy filter and the time-discrete Kalman filter for the current about 5 A and for the angular velocity about 0.2 rad s^{-1} , but the dynamics of the filters are the same. Discretizing the Kalman-Bucy filter by replacing in Simulink the continuous integrator with an Euler integrator and a sample frequency of 100 Hz, the result is visible in figure 2.13. In the stationary case the results are the same as for the discrete Kalman filter, but the dynamics are different. The benefit of the Kalman-Bucy filter is, no matrix inversion must be calculated which speeds up the filter velocity and increases the accuracy, because inverting matrices of small or big values lead to rounding errors[35]. The drawback is, just one sample time can be used, but the sensors have different sample times which is described in section 3.3.1.

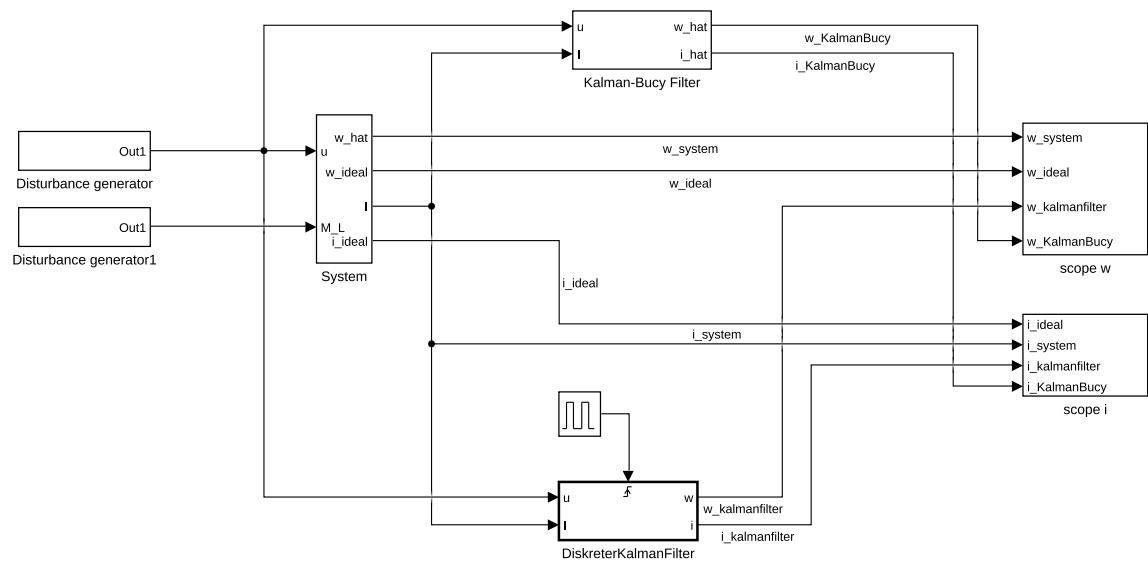


Figure 2.8: Comparsion of a discrete Kalman filter with a time-continuous Kalman-Bucy filter.

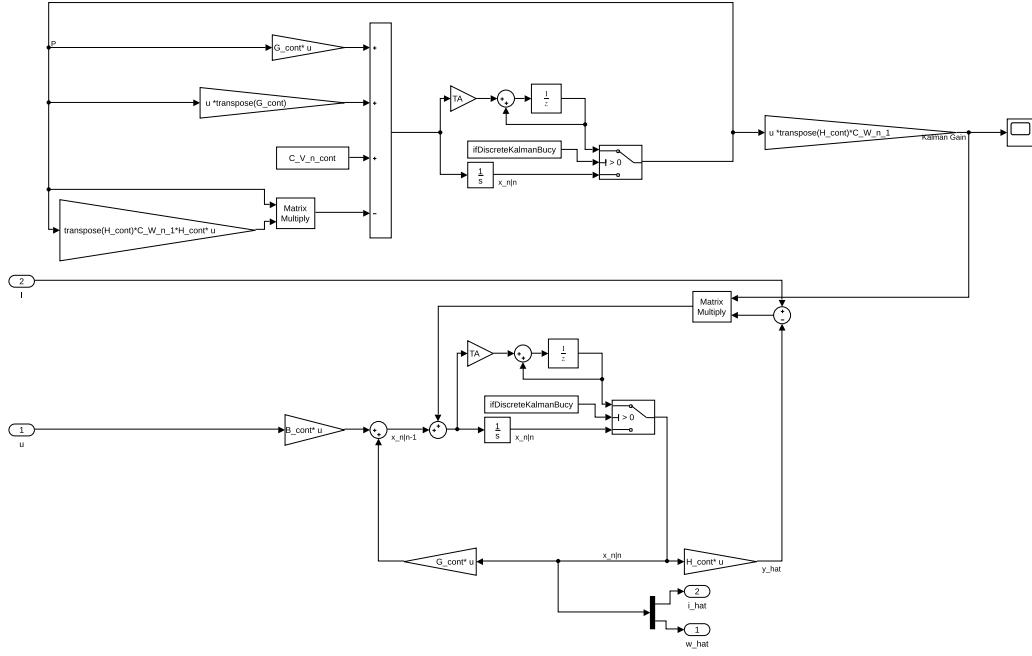


Figure 2.9: Blockdiagram of the Kalman-Bucy filter[34].

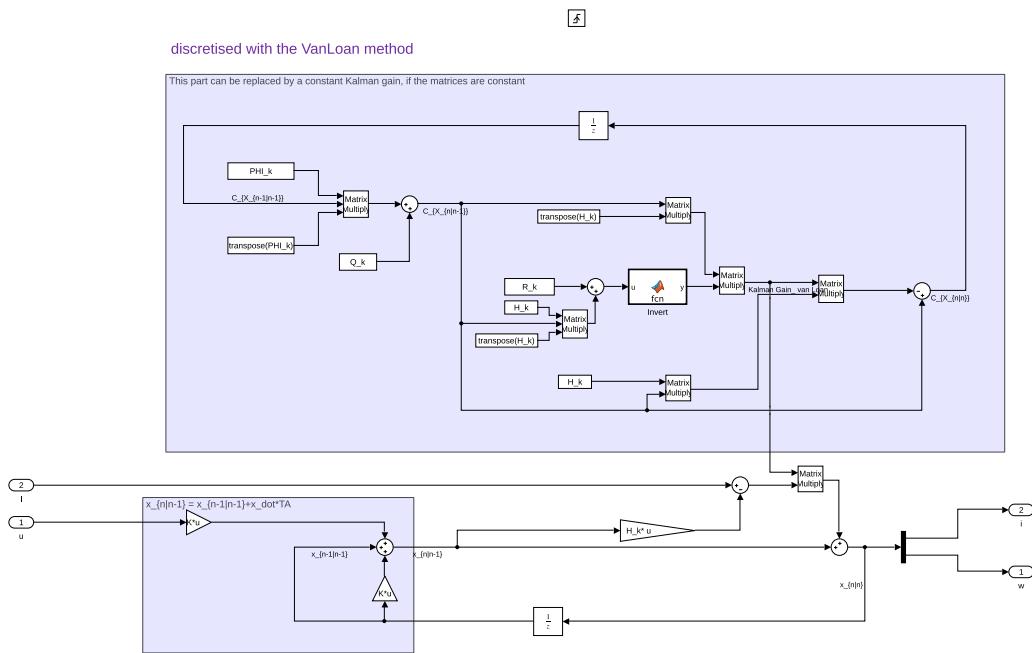


Figure 2.10: Blockdiagram of the discrete Kalman filter.

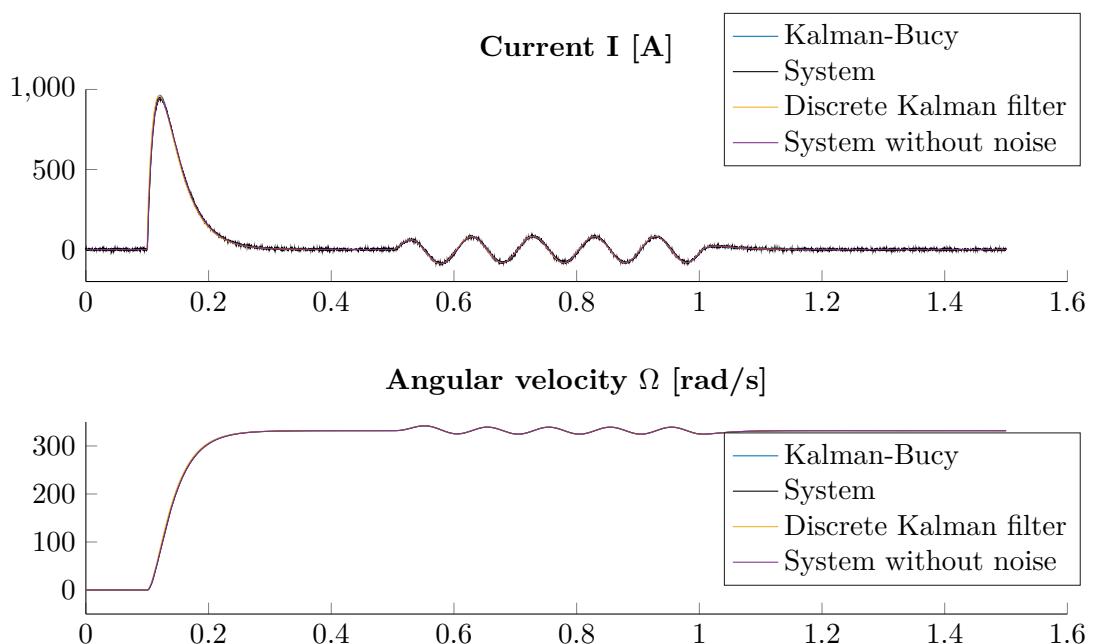


Figure 2.11: Filter dynamics comparsion.

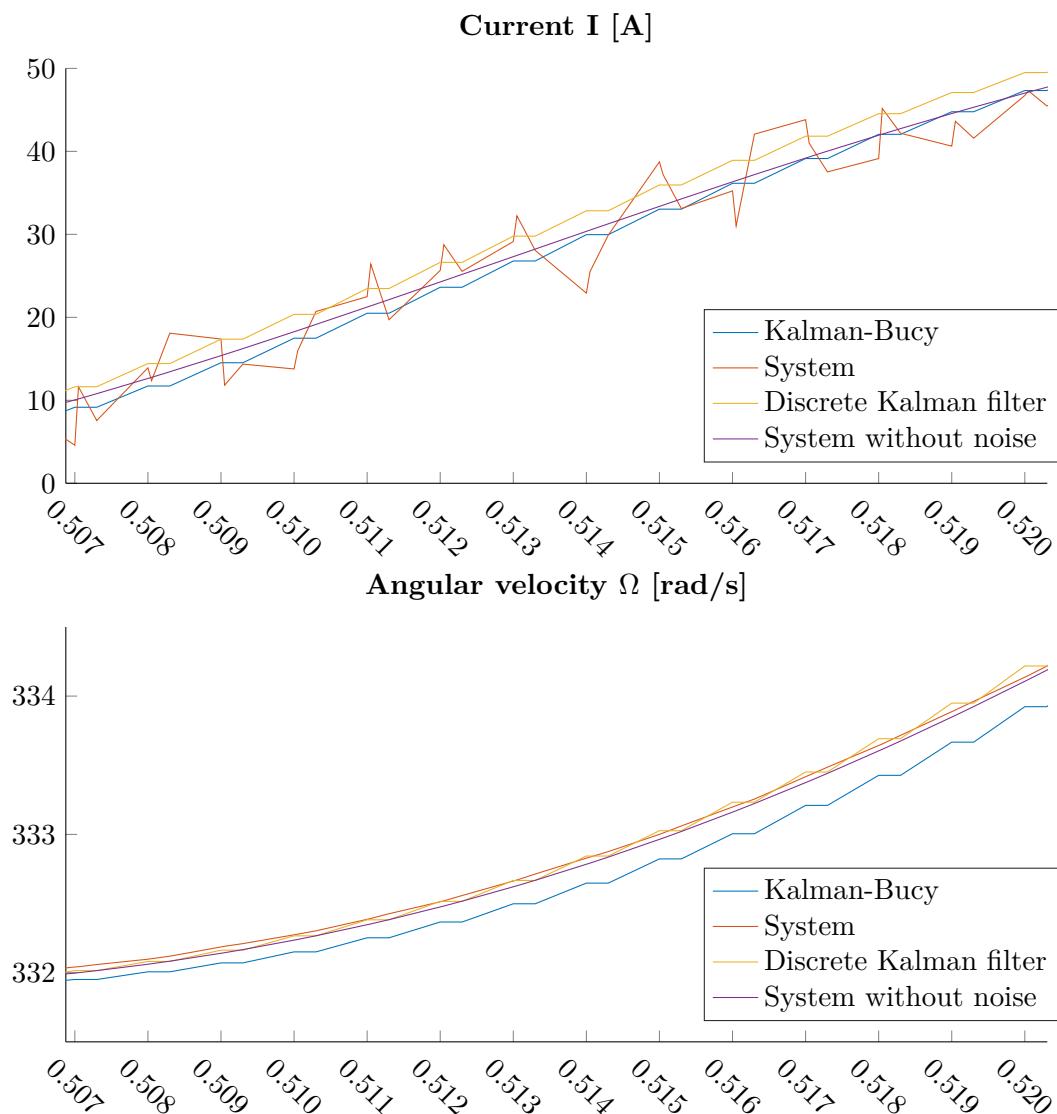


Figure 2.12: Zoom in of figure 2.11.

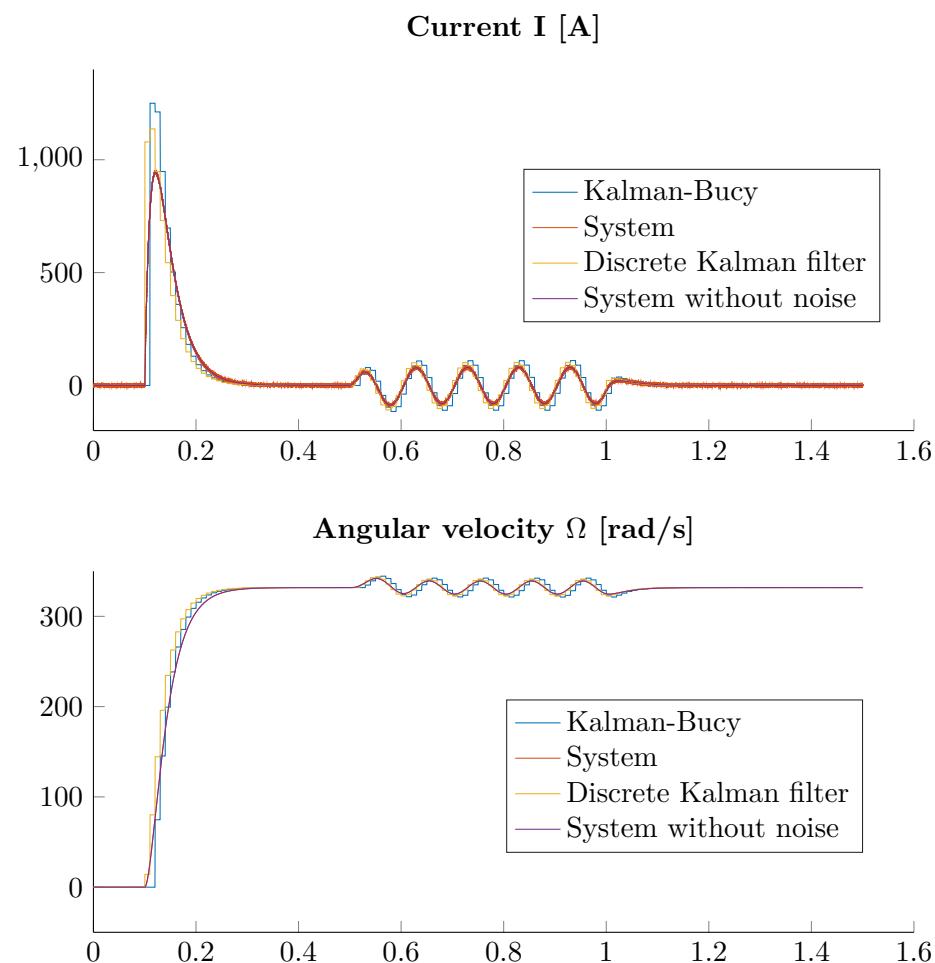


Figure 2.13: Comparsion of the discretised Kalman filter with the discretised Kalman-Bucy filter at a sample frequency of 100 Hz.

Extended Kalman Filter (EKF)

The EKF is a derivation from the linear Kalman filter to use the filter for nonlinear systems by linearizing the system. On considers a system

$$\mathbf{X}_n = \mathbf{g}(\mathbf{X}_{n-1|n-1}, \mathbf{B}_n) + \mathbf{V}_n \quad (2.94)$$

$$\mathbf{Y}_n = \mathbf{h}(\mathbf{X}_n) + \mathbf{W}_n \quad (2.95)$$

with the Taylor Series of $\mathbf{g}()$ about the point $x_{n|n-1}$ [36]

$$\mathbf{g}(\mathbf{X}_{n-1|n-1}, \mathbf{B}_n) = \mathbf{g}(\hat{\mathbf{X}}_{n-1|n-1}, \mathbf{B}_n) + J_g(\hat{\mathbf{X}}_{n-1|n-1}, \mathbf{B}_n)(x_{n-1|n-1} - \hat{x}_{n-1|n-1}) + H.O.T \quad (2.96)$$

with H.O.T the neglected higher order terms[36] and J_g is the jacobian matrix defined as

$$J_g := \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \dots & \frac{\partial g_1}{\partial x_n} \\ \vdots & \ddots & & \vdots \\ \frac{\partial g_n}{\partial x_1} & \frac{\partial g_n}{\partial x_2} & \dots & \frac{\partial g_n}{\partial x_n} \end{bmatrix} \quad (2.97)$$

The mean of equation (2.94) with respect to all measurements until $\mathbf{y}_{(n-1)}$ can be calculated with (2.96)

$$\hat{x}_{n|n-1} = E[\mathbf{X}_n | \mathbf{Y}_{(n-1)} = \mathbf{y}_{(n-1)}] = E[\mathbf{g}(\mathbf{X}_{n-1|n-1}, \mathbf{B}_n) + \mathbf{V}_n] = \mathbf{g}(\hat{\mathbf{X}}_{n-1|n-1}, \mathbf{B}_n). \quad (2.98)$$

The covariance matrix $\mathbf{C}_{(n|n-1)}$ is calculated as

$$\mathbf{C}_{\mathbf{X}_{(n|n-1)}} = J_g(\hat{\mathbf{X}}_{n-1|n-1}, \mathbf{B}_n) + \mathbf{C}_{\mathbf{V}_n}. \quad (2.99)$$

Due to the linearization, the covariance matrix $\mathbf{C}_{\mathbf{X}_{(n|n-1)}}$ is not constant like in the linear case and must be evaluated for every step of the filter. For Calculating $\hat{x}_{n|n}$, in equation (2.78) the term $\mathbf{H}_n \hat{\mathbf{x}}_{n|n-1}$ is replaced by the mean of \mathbf{y}_n

$$E[\mathbf{y}_n] = h(\hat{x}_{n-1|n-1}). \quad (2.100)$$

The linear part of \mathbf{y}_n , \mathbf{H}_n , can be calculated with the Jacobian matrix J_h (2.97).

Multiplikative extended Kalman filter (MEKF)

The MEKF is an extension of the EKF for attitude estimation, which uses unit quaternions as base for the attitude. The attitude is determined by three angles which means the covariance matrix has $\text{rang}(\mathbf{C}_{V_n}) = 3$. Using quaternions to represent the attitude the covariance matrix is of size 4×4 but the rang remains three[12], which results in a singularity when trying to invert the matrix. When implementing a quaternion based Kalman filter, it is assumed that the rang of the matrix is four. To reduce the number of states for the attitude and getting a process covariance matrix of size 3×3 , not the unit quaternion itself, but the error between the real attitude and the estimated attitude is used as a state. The error quaternion $\delta \mathbf{q}$ is defined as the rotation from the estimated attitude to the real attitude

$$\mathbf{q} = \hat{\mathbf{q}} \otimes \delta \mathbf{q}. \quad (2.101)$$

The multiplikation of two unit quaternions is used, because adding two quaternions do not result anymore in a unit quaternion and the resulting quaternion must be normalized [11]. The derivation of the attitude error unit quaternion is

$$\delta \dot{\mathbf{q}} = \hat{\mathbf{q}}^{-1} \otimes \dot{\mathbf{q}} + \dot{\hat{\mathbf{q}}}^{-1} \otimes \mathbf{q} \quad (2.102)$$

with the inverse of the estimated attitude as

$$\dot{\hat{\mathbf{q}}}^{-1} = -\frac{1}{2} \begin{bmatrix} 0 \\ \hat{\omega}_b \end{bmatrix} \otimes \hat{\mathbf{q}}^{-1}. \quad (2.103)$$

The error attitude can be simplified according to [11] to

$$\delta\dot{\mathbf{q}} = \begin{bmatrix} 0 \\ -\hat{\omega} \times \delta\mathbf{q}_v \end{bmatrix} + \frac{1}{2} \begin{bmatrix} -\delta\mathbf{q}_v \cdot \delta\hat{\omega} \\ \delta\hat{\omega} + \delta\mathbf{q}_v \times \delta\hat{\omega} \end{bmatrix}. \quad (2.104)$$

Due to the assumption of a small error between the estimated attitude and the real attitude, like already said in (2.22), the error quaternion is almost an identity quaternion

$$\delta\mathbf{q} \approx \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (2.105)$$

In this case the dynamic of the scalar is omitted and only the vector part is relevant. The rotation matrix simplifies to

$$\mathbf{R}_e^b(\delta\mathbf{q}) \approx [\mathbf{I} - 2[\delta\mathbf{q}_v \times]]. \quad (2.106)$$

The quaternion error state $\delta\mathbf{q}$ is replaced by a vector of small angles $\boldsymbol{\alpha}$. There exist multiple representations of this vector[12]. In this thesis $\boldsymbol{\alpha}$ is defined as in [11] as

$$\boldsymbol{\alpha} = 2\delta\mathbf{q}_v. \quad (2.107)$$

The first benefit of this representation is, in equation (2.104) $\frac{1}{2}$ is omitted. The second advantage is the simplification of the calculation of the new estimated attitude, which is shown later. The derivative of this small angle vector is calculated according to[11] to

$$\dot{\boldsymbol{\alpha}} = -\hat{\omega} \times \boldsymbol{\alpha} - \mathbf{b}_w. \quad (2.108)$$

The rotation matrix can now be expressed in terms of the error angle $\boldsymbol{\alpha}$ and the estimated attitude $\hat{\mathbf{q}}$ from the previous step as

$$\mathbf{R}_e^b(\mathbf{q}) \approx [\mathbf{I} - \boldsymbol{\alpha} \times] \mathbf{R}_e^b(\hat{\mathbf{q}}) \quad (2.109)$$

and

$$\mathbf{R}_b^e(\mathbf{q}) \approx \mathbf{R}_b^e(\hat{\mathbf{q}}) [\mathbf{I} - \boldsymbol{\alpha} \times]. \quad (2.110)$$

The estimated measurements must be calculated using equation (2.109) and (2.110) with the state $\boldsymbol{\alpha}$. After the measurement step of the EKF, the new attitude is calculated from the previous estimated attitude and the small error angle. Due to the assumption, the small error is twice the vector part of the error quaternion, the new attitude can be calculated by calculating an unnormalized quaternion with

$$\mathbf{p} = \hat{\mathbf{q}}_{k-1} \otimes \begin{bmatrix} 2 \\ \boldsymbol{\alpha} \end{bmatrix} \quad (2.111)$$

and then renormalizing

$$\hat{\mathbf{q}}_k = \frac{\mathbf{p}}{\|\mathbf{p}\|}. \quad (2.112)$$

The mean of the attitude error angle is zero, which means, the small error angle must be set to zero after the predictor step. This is different to the papers [11, 13, 12], because they only use errors between the real value and the estimated value as states. In this thesis, only for the attitude an error value is used as a state. For other states like the velocity or the position the velocity/position itself is the state and these states must be predicted in the prediction step of the EKF. The following table summarizes the steps of the MEKF

1	calculate $\hat{\mathbf{q}}_{k+1}$ with equation (2.26)
2	EKF prediction state using the equations defined in section 2.4.1
3	reset all error states to zero
4	EKF measurement step
5	update $\hat{\mathbf{q}}_{k+1}$ with the error state $\boldsymbol{\alpha}$ with equations (2.111) and (2.112)

Table 2.1: Steps of the MEKF.

2.4.2 Mahony explicit complementary filter

Mahony et al. presented in [28] and [37] a new nonlinear filter for estimating the attitude of an aerial vehicle. This filter does not use any further information than the gyroscope, accelerometer and magnetometer sensor data. Figure 2.14 shows the blockdiagram of the Mahony passive complementary filter. The filter was named complementary filter, because it has the same structure than a linear complementary filter. \mathbf{R} is the estimated rotation matrix calculated from the accelerometer and the magnetometer measurements. $\boldsymbol{\omega}_y$ are the gyroscope sensor data. k is a gain to parameterize the Filter, to adjust the performance to the desired needs. Due to not linearizing the equations, in [28] was shown, that the filter is globally asymptotic stable. The filter can be described as

$$\dot{\hat{\mathbf{R}}} = (\mathbf{R}_y \boldsymbol{\omega}_y + k_P \hat{\mathbf{R}} \boldsymbol{\omega}) \times \hat{\mathbf{R}} \quad (2.113)$$

with

$$\boldsymbol{\omega} = \text{vex}(\mathbb{P}_a(\tilde{\mathbf{R}})) \quad (2.114)$$

with the error matrix between the real rotationmatrix and the estimated rotation matrix

$$\tilde{\mathbf{R}} = \hat{\mathbf{R}}^T \mathbf{R}. \quad (2.115)$$

To avoid the need to calculate the rotation matrix \mathbf{R}_y the filter was evolved to directly use the magnetometer and accelerometer input. For this the correction term (2.114) was changed to

$$\boldsymbol{\omega} = \sum_{i=1}^2 k_i \mathbf{v}_i \times \hat{\mathbf{v}}_i \quad (2.116)$$

with \mathbf{v}_i the magnetometer and accelerometer input and $\hat{\mathbf{v}}_i$ the estimated vector from the estimated attitude. This filter was named explicit complementary filter[28]. In practice this filter is quite computational expensive, because from the rotation matrix the attitude must be calculated. Due to this disadvantage the filter was evolved to use directly unit quaternions[28]. The quaternion differential equation is evolved to

$$\dot{\hat{\mathbf{q}}} = \frac{1}{2} \hat{\mathbf{q}} \otimes \mathbf{p} (\boldsymbol{\omega}_y - \hat{\mathbf{b}}_\omega + k_P \boldsymbol{\omega}_{\text{meas}}) \quad (2.117)$$

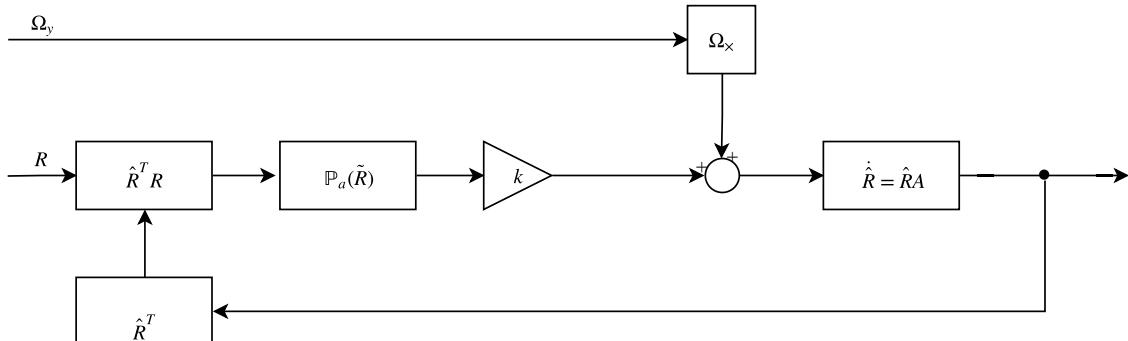


Figure 2.14: Blockdiagram of the Mahony passive complementary filter.

with ω_{meas} as

$$\omega_{meas} = -vex \left(\sum_{i=1}^2 \frac{k_i}{2} \left(\mathbf{v}_i \hat{\mathbf{v}}_i^T - \hat{\mathbf{v}}_i \mathbf{v}_i^T \right) \right) \quad (2.118)$$

and the gyro bias $\dot{\hat{\mathbf{b}}}_\omega$ differential equation defined as

$$\dot{\hat{\mathbf{b}}} = -k_I \omega_{meas}. \quad (2.119)$$

This type of filter needs much less computational power than a kalman filter, because it needs no large matrices multiplication and no inversion of any matrix. The parameter k_P , k_I , k_1 and k_2 can be optimized to offer the desired performance. This optimization is described in section 3.1.4.

2.4.3 Comparing filter to already existing Attitude estimator Pixhawk 4

In this section the state estimator of flight controller Pixhawk 4 seen in figure 2.15 is examined[38]. This hardware includes several sensors like accelerometer, magnetometer, barometer and can be extended with different sensors like a GPS module, wind velocity sensor or range sensors. As estimator Pixhawk 4 uses a quaternion based EKF with the states

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \mathbf{v} \\ \mathbf{p} \\ \mathbf{b}_q \\ \mathbf{b}_v \\ mag_N \\ mag_E \\ mag_D \\ mag_X \\ mag_Y \\ mag_Z \\ vWN \\ vWE \end{bmatrix} \quad (2.120)$$

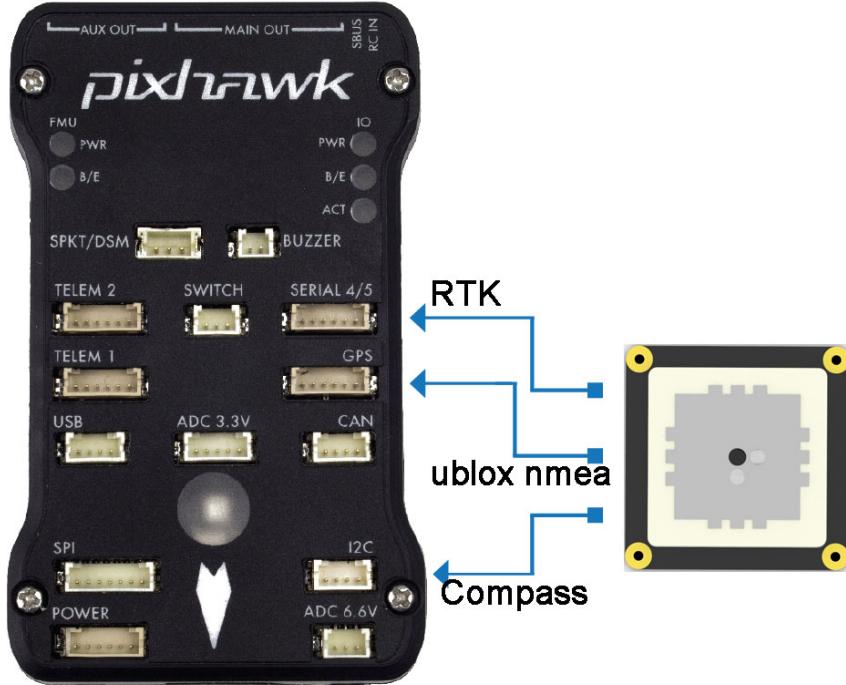
with \mathbf{q} the attitude, \mathbf{v} the velocity in NED direction, \mathbf{p} the position in the NED frame, mag_x , $x \in \{N, E, D\}$ the magnetic field in the inertia NED frame, mag_y , $y \in \{X, Y, Z\}$ the magnetic field in the body frame. The magnetic field in the body frame is used online to determine the hard iron disturbance described in 2.2.3[39] as

$$\mathbf{m}_{b_{meas}} = \mathbf{R}_e^b(\hat{\mathbf{q}}) \begin{bmatrix} mag_N \\ mag_E \\ mag_D \end{bmatrix} + \begin{bmatrix} mag_X \\ mag_Y \\ mag_Z \end{bmatrix}. \quad (2.121)$$

The accelerometer and the gyroscope are used in the prediction step of the Kalman filter like the INS filters in this thesis. The measurements are fused in multiple steps in the correction step, therefore the filter can also estimate if one or more sensors are missing or additional sensors are added. The measurements for the position and the velocity come from GPS module and the barometer and are handled same like described in the INS filters in chapter 3. An additional wind sensor can be used to determine the true air speed

$$v_{TAS} = \sqrt{(v_N - vWN)^2 + (v_E - vWE)^2 + v_D^2} \quad (2.122)$$

to improve the estimation of the inertial velocity. If the error between the measured value and the predicted value is higher than a multiple of the standard deviation, the measurement step is skipped[41]. While starting, until a height of 1.5 m, the magnetometer values are used



Source: [40]

Figure 2.15: Pixhawk 4 hardware.

with the accelerometer values to estimate the yaw angle. This value is then compared with the estimated yaw angle. This method is more robust to large start-up gyro biases[42], but has the drawback, when the accelerometer is failing, no angle estimation is available, because the yaw angle calculation needs the accelerometer. When the flight controller is above 1.5 m, the magnetometer values itself are used as measurement, which is more precise due to the three-dimensional vector instead of using a single angle. If no position/velocity sensor is available, an AHRSEKF is used to estimate the attitude.

Chapter 3

Simulations

The input data for the filter was simulated by F. Bauer. The trajectory was chosen to convert as much wind energy as possible into electrical energy. The position and the velocity of the simulated kite can be seen in figure 3.1. From 80 s on the kite is in cross wind flight and flies a periodic trajectory and produces electrical energy. This can be read out from the third diagram of figure 3.2. From this time on, there is a high acceleration part which is measured from the kite accelerometer. In figure 3.2 the attitude of the kite represented by three Euler angles is shown. The variances used in the Kalman filter have the values from table 3.1 and 3.2. These values are used in the Pixhawk 4 firmware[43]. The gyroscope bias, magnetometer, barometer and GPS variances are used from Jeffrey D. Barton[8]. The variance for the Euler

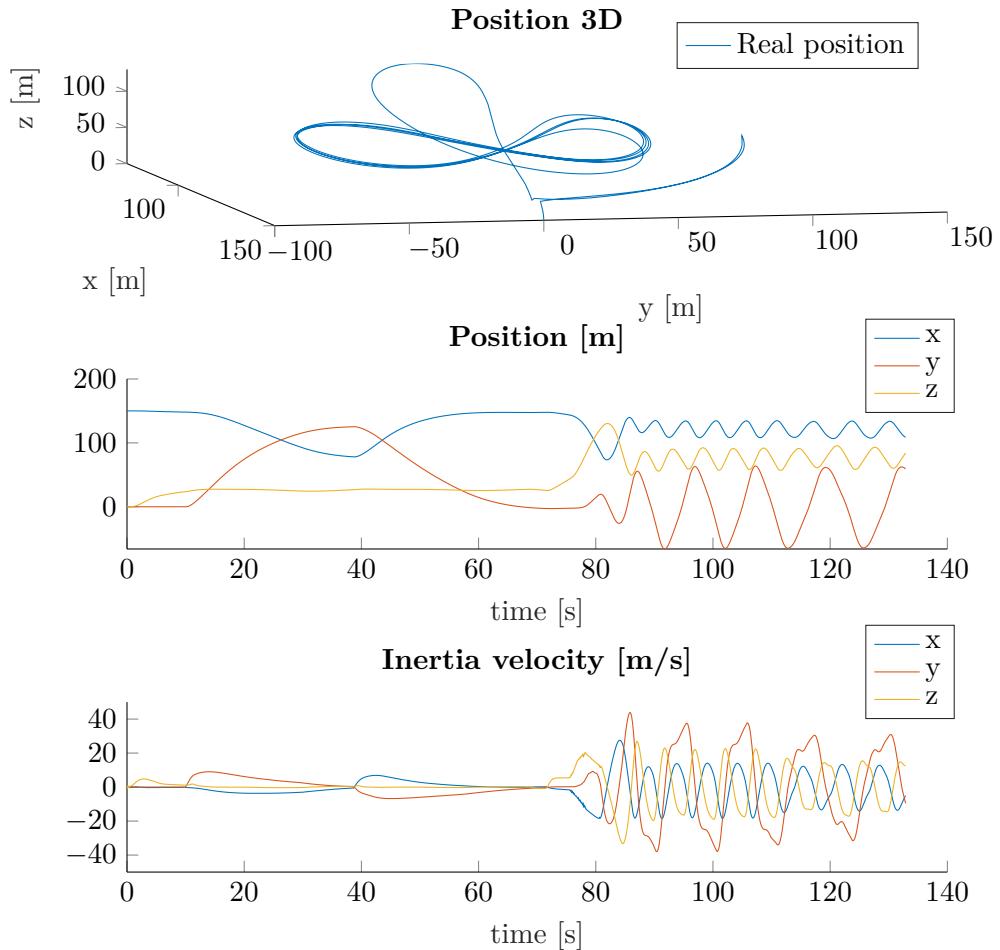


Figure 3.1: Position and velocity of the kite.

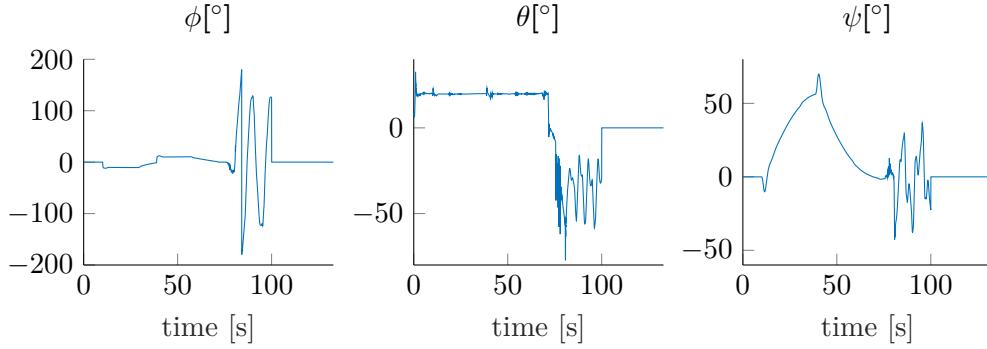


Figure 3.2: Attitude of the kite.

Sensor	Variance
Gyroscope	$2.5 \times 10^{-4} \text{ rad s}^{-1}$
Accelerometer	0.1225 m s^{-2}
Magnetometer	$3.24 \times 10^{-2} \mu\text{T}$
GPS Position	10 m
GPS Velocity	1.5 m s^{-1}
Barometer	0.66 m
Line length	0.3 m
Line angle sensor	3°

Table 3.1: Sensor precision.

angles and the quaternion states come from the gyroscope due to the equations (2.25) and (2.7).

State variable	Variance σ	Annotation
Euler angles γ	$2.5 \times 10^{-4} \text{ rad s}^{-1}$	Comes from the gyroscope
quaternion q	$2.5 \times 10^{-4} \text{ rad s}^{-1}$	Comes from the gyroscope
Gyroscope bias b_w	$1 \times 10^{-6} \text{ rad s}^{-1}$	Used from [8]
Attitude error α	$2.26 \times 10^{-4} \text{ rad s}^{-1}$	Gyroscope plus Gyroscope bias

Table 3.2: State precision.

In the next sections some filters are presented and evaluated. To evaluate the errors between the real value and the estimated value, the root-mean-square deviation (RMSD) function is used

$$\text{rmsd}(y) = \sqrt{\frac{\sum_{t=1}^T (\hat{y}_t - y_t)^2}{T}}. \quad (3.1)$$

In all of this filter the following assumptions are used

Assumption 5. *The gyroscope bias is constant or slowly varying[37].*

$$\dot{b}_w \approx 0 \quad (3.2)$$

Assumption 6. *The velocity of the kite in body coordinates v_b can be calculated from the inertia velocity as*

$$v_b = \begin{bmatrix} ||v_e|| \\ 0 \\ 0 \end{bmatrix} \quad (3.3)$$

because it is assumed, that the kite flies always in the direction of the body x axis.

It's not possible to find the variance of p_{tether} algebraically due to the nonlinearity. In this case the variance can be calculated by linearizing the position and then calculating the variance in a single point or by creating a simulation of the position at specific angles and tether length and calculating the variance of all of these points. In this work it's assumed, that the variance, estimated with a simulation at a specific angle is the maximum variance which can be found.

Assumption 7. *The variance of the tether angle position estimation is constant.*

In the first part of this chapter until section 3.3.3 it's also assumed

Assumption 8. *The tether is a straight line whereas the position can be calculated as in 2.2.6.*

3.1 AHRS filter

3.1.1 Euler angle based AHRS EKF

The AHRS EKF based on Euler angles as states is the simplest filter in this thesis. The attitude is directly visible in the states, which makes debugging and checking the filter quite easy. The drawback is, as already described in 2.1.3 a gimbal lock can occur and the filter can fail. As states the attitude and the biases $\mathbf{b}_w \in \mathbb{R}^3$ of the angular velocity are used. The biases are used, because they are important for integrating the angular velocity to get the attitude. As measurements, the accelerometer and the magnetometer are used. The angular rate is used in the update step to estimate the new attitude. The GPS velocity is used to compensate the centripetal acceleration. The state equation is defined as

$$\begin{bmatrix} \dot{\gamma} \\ \dot{\mathbf{b}}_w \end{bmatrix} = \begin{bmatrix} \mathbf{S}(\omega_b - \mathbf{b}_w) \\ \mathbf{0} \end{bmatrix} \quad (3.4)$$

with γ the attitude and \mathbf{b}_ω the gyroscope bias. The measurement equation is expressed as

$$\begin{bmatrix} \mathbf{a}_b \\ \mathbf{m}_b \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{R}}_e^b(\gamma) \mathbf{g}_e + (\omega_b - \mathbf{b}_w) \times \mathbf{v}_b \\ \hat{\mathbf{R}}_e^b \mathbf{m}_e \end{bmatrix} \quad (3.5)$$

with the estimated rotation matrix $\hat{\mathbf{R}}_e^b$. Comparing the upper part of the right hand side of equation (3.5) and (2.30) which is

$$\mathbf{a}_b = \dot{\mathbf{v}}_b + \mathbf{w}_b \times \mathbf{v}_b - \mathbf{g}_b, \quad (3.6)$$

it is visible, that the first part of the equation, the translational acceleration, is missing. Because of this, at high translational accelerations high errors in the attitude exist which is visible in figure 3.3. Due to the low sample rate of the GPS module and the fact that differentiating noisy signals is not a good choice, it's impossible to estimate the translational acceleration. At low translational accelerations like at the start phase of the kite until about 80 s the error is low and the estimated gyro bias converges to the real gyro bias seeing in figure 3.3 and 3.4. The position is estimated from the GPS module. As seen in figure 3.3, the filter does not touch the angles $\pm 90^\circ$ for the angle θ , for what reason no singularities occurred. Seeing in figure 3.4, at high accelerations, the estimation of the gyro bias is completely failing.

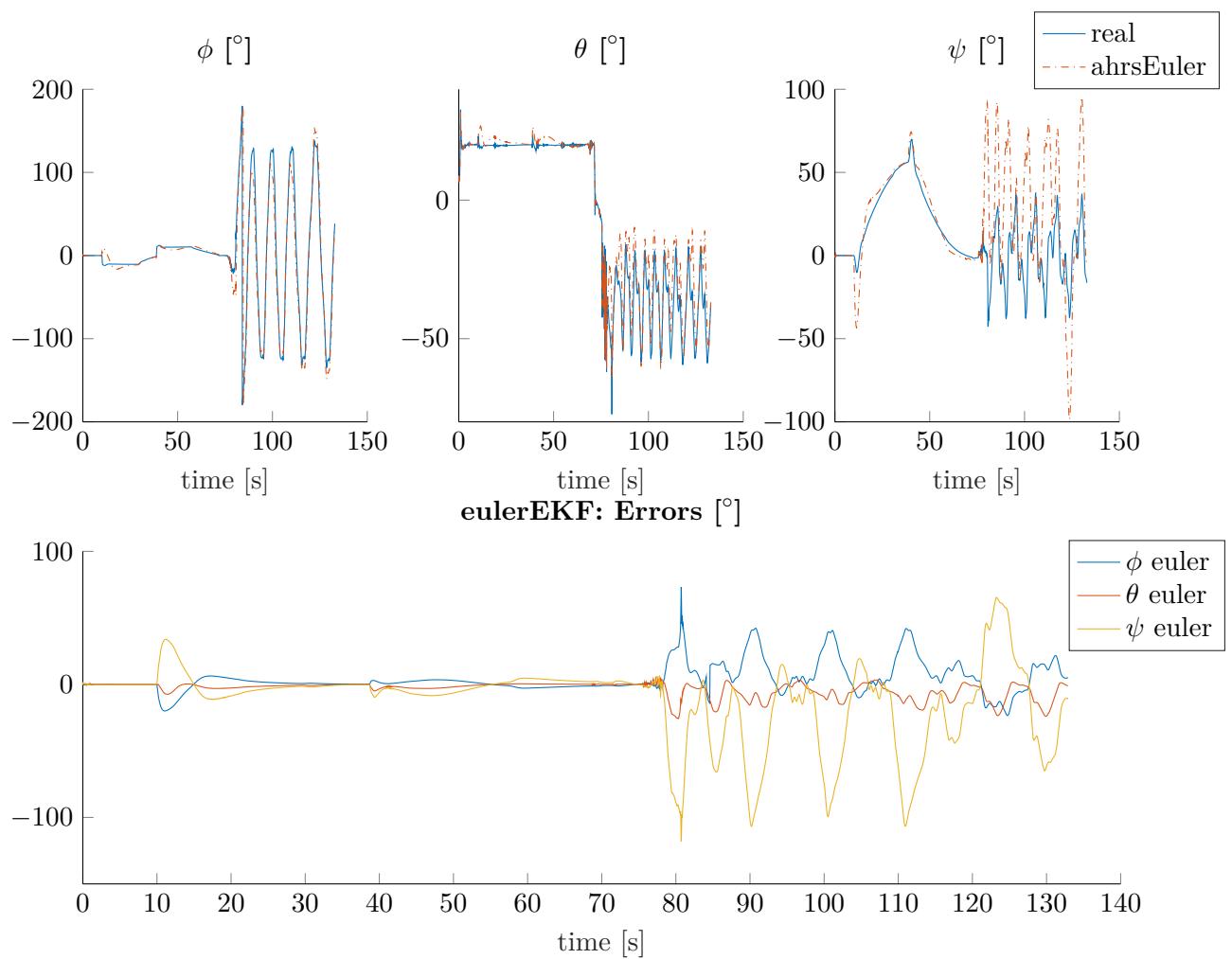


Figure 3.3: Simulation result of the Euler AHRS EKF.

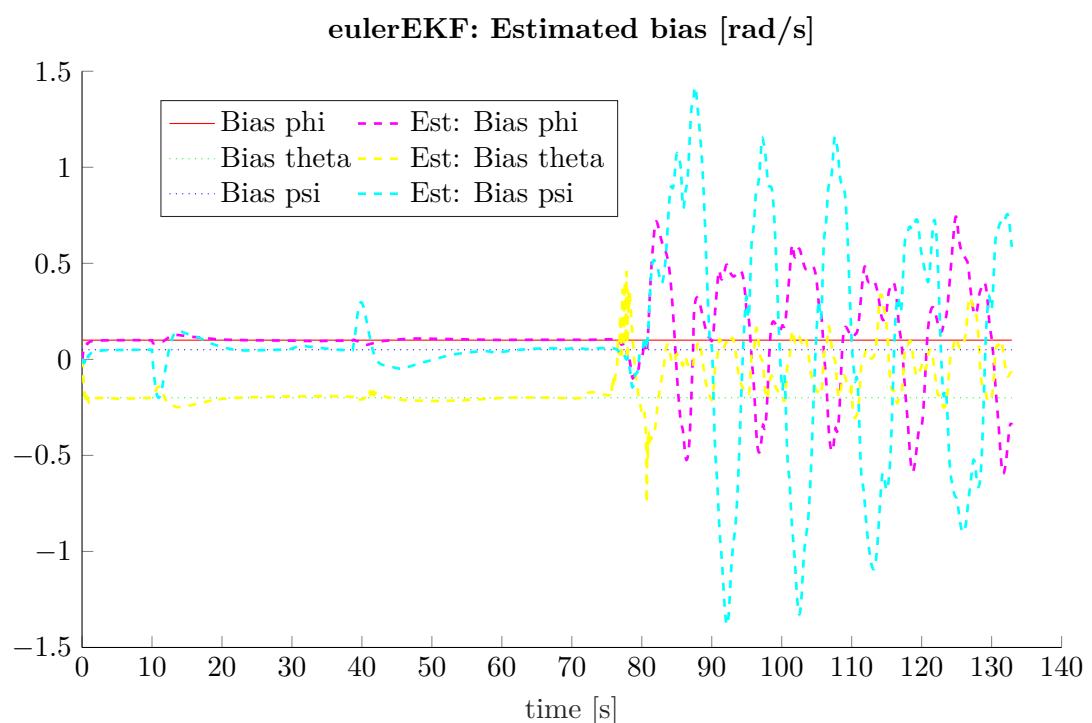


Figure 3.4: Estimated gyroscope bias. Due to the high translational acceleration, the gyro bias and the attitude have high errors.

3.1.2 Quaternion based AHRS EKF

The AHRS EKF based on quaternions has now four states for the representation of the attitude, but does not have the problem singularities occur like for the Euler based EKF. The state equation is defined as

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{b}_w} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\mathbf{q} & 0 \\ \omega_b - \mathbf{b}_w & \mathbf{0} \end{bmatrix}. \quad (3.7)$$

The measurement equations are defined similar to the Euler based AHRS EKF with the difference, equation (2.19) is used to rotate the gravity / magnetic field vector into the body frame

$$\mathbf{y} = \begin{bmatrix} \mathbf{a}_b \\ \mathbf{m}_b \end{bmatrix} = \begin{bmatrix} R_e^b(\hat{\mathbf{q}})\mathbf{g}_e + (\omega_b - \mathbf{b}_w) \times \mathbf{v}_b \\ R_e^b(\hat{\mathbf{q}})\mathbf{m}_e \end{bmatrix}. \quad (3.8)$$

The derivative of the quaternion was already defined in equation (2.25). Comparing the estimated angles of the Euler based AHRS EKF from 3.1.1 in figure 3.3 it is visible, that the characteristic of the filters are the same. Like the Euler AHRS EKF, the quaternion AHRS EKF does not estimate well, when high translational acceleration is present.

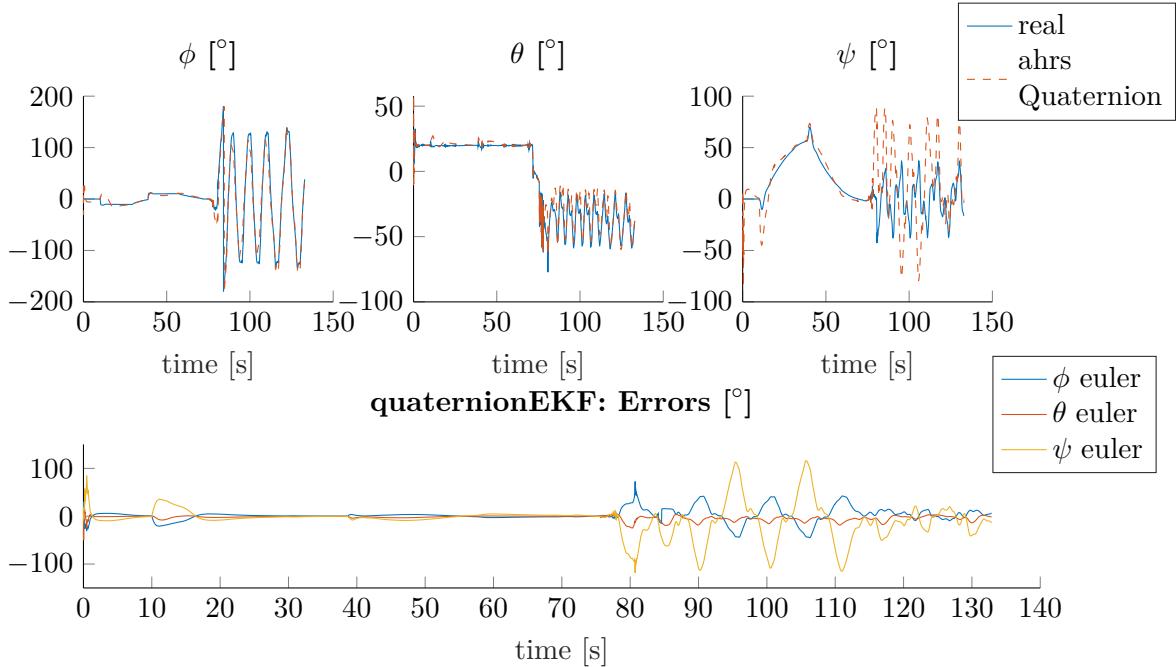


Figure 3.5: Quaternion based AHRS EKF.

3.1.3 AHRS MEKF

Following the description in 2.4.1 a MEKF was developed to estimate the attitude of the kite. The system and measurement equations are defined similar to the Euler and quaternion EKF

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\boldsymbol{\alpha}} \\ \dot{\mathbf{b}_w} \end{bmatrix} = \begin{bmatrix} -(\boldsymbol{\omega} - \mathbf{b}_w) \times & -\mathbf{I}_3 \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \mathbf{b}_w \end{bmatrix} \quad (3.9)$$

$$\mathbf{y} = \begin{bmatrix} \mathbf{a}_b \\ \mathbf{m}_b \end{bmatrix} = \begin{bmatrix} (\mathbf{I}_3 - \boldsymbol{\alpha} \times) R_e^b(\hat{\mathbf{q}}) \mathbf{g}_e + (\boldsymbol{\omega} - \mathbf{b}_w) \mathbf{v}_b \\ (\mathbf{I}_3 - \boldsymbol{\alpha} \times) R_e^b(\hat{\mathbf{q}}) \mathbf{m}_e \end{bmatrix} \quad (3.10)$$

with $\boldsymbol{\alpha}$ the small error angle vector. The result of this filter is shown in figure 3.6. The result

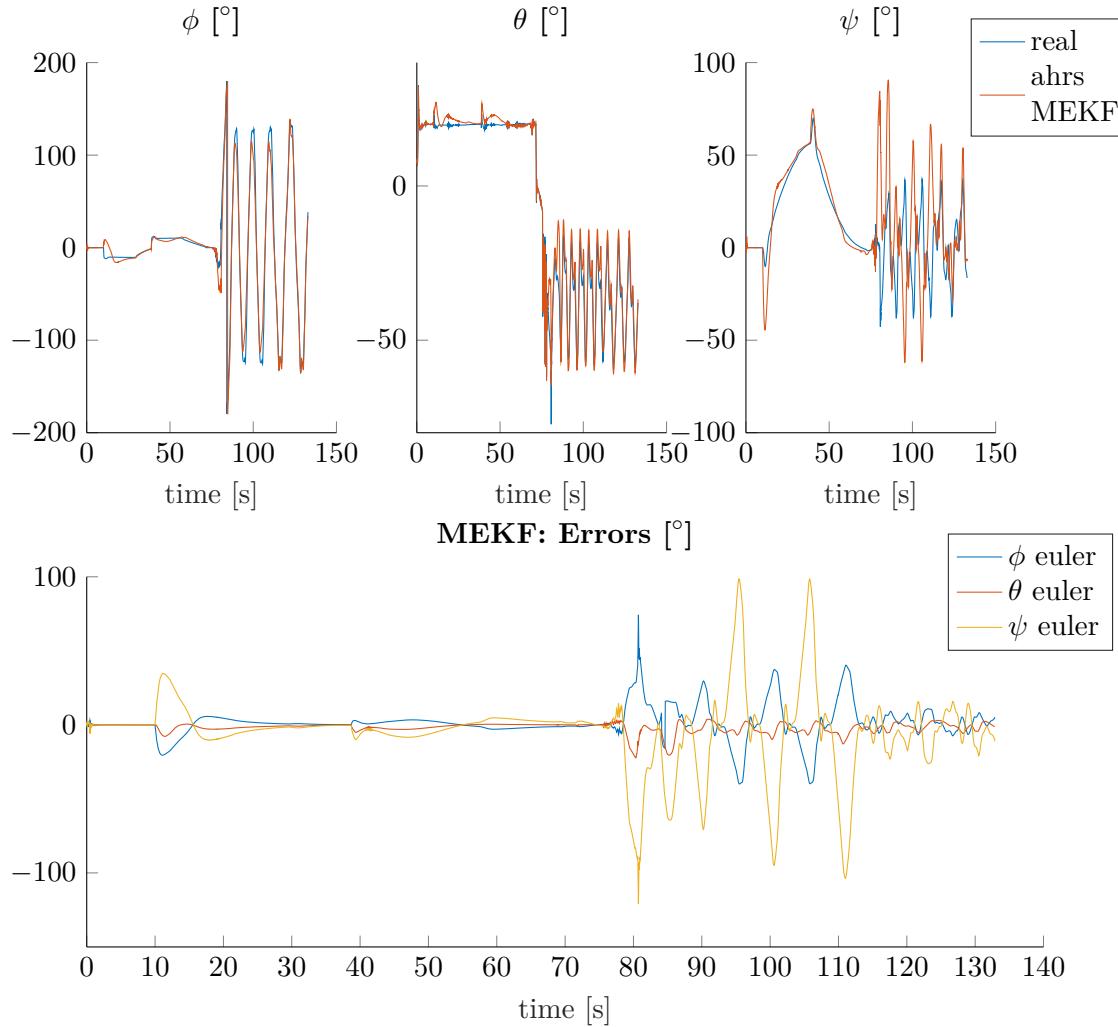


Figure 3.6: Attitude estimation of an AHRS MEKF.

is similar to the results from the other AHRS filters, but with the advantage to not have any singularities like the quaternion AHRS EKF and having only three states (α) for the attitude which increases the performance of the filter.

3.1.4 Explicit complementary filter

Seeing in figure 3.7, the filter is in the starting phase quite good and similar to the EKF filter, but when high translational acceleration is present, the filter is unusable. This filter cannot be extended to be an INS filter. The parameters of the filter are determined by optimizing the filter on this trajectory, with the weight for the importance of the error angle as described in table 3.3. The weight for the yaw (ψ) angle is choose lower than the roll(ϕ) and pitch (θ)

Weight	ϕ	θ	ψ
	1	1	0.5

Table 3.3: Weights of the explicit complementary filter optimization.

angle, because it's more important of being stable in x and y axis.

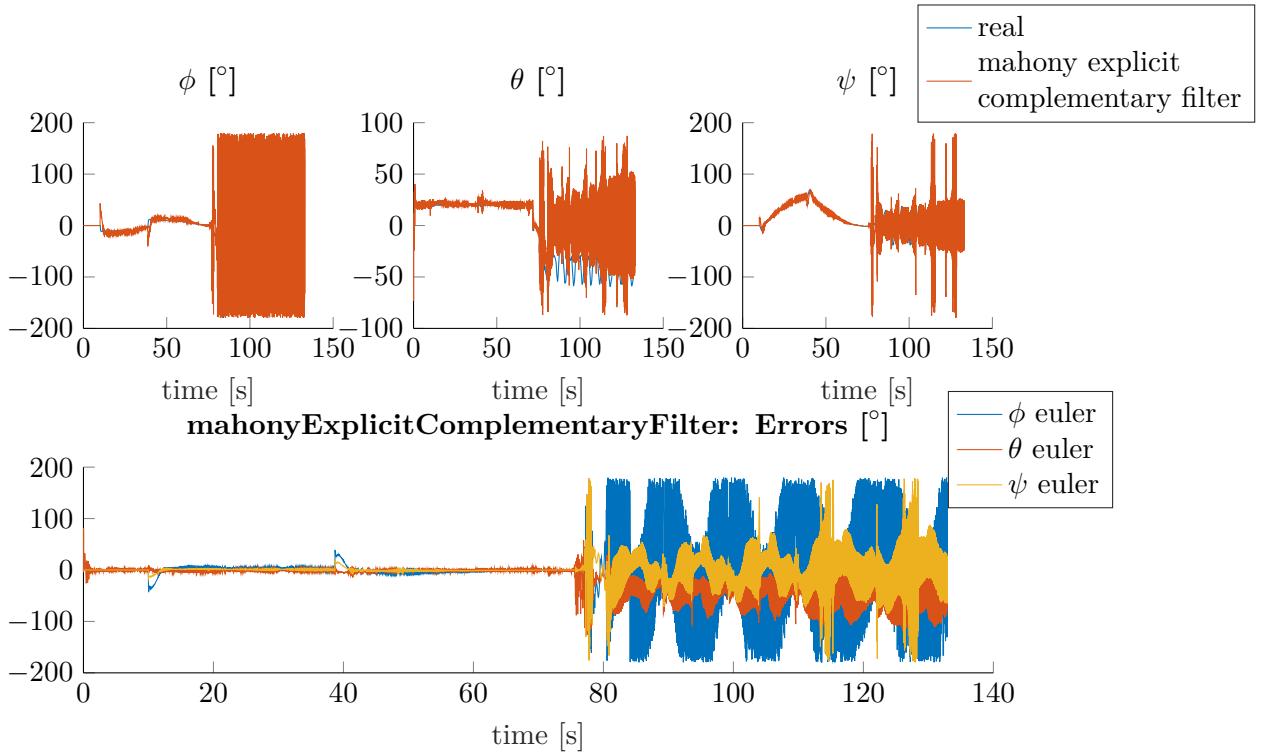


Figure 3.7: Attitude estimation of the explicit complementary filter.

3.2 INS filter

To avoid the problem of the filters at high translational accelerations, an INS filter can be used. This filter uses the velocity as additional state, whereas the estimated acceleration is not anymore used in the correction step, but the measured acceleration is rotated into the earth frame and used in the state equation of the velocity instead of the measurement equations. The following equations use the MEKF approach described in section 2.4.1 but the INS filter can also be described in Euler or quaternion representation. The state equation is defined as

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{\boldsymbol{\alpha}} \\ \dot{\boldsymbol{p}} \\ \dot{\boldsymbol{v}} \\ \dot{\boldsymbol{b}_w} \end{bmatrix} = \begin{bmatrix} -(\boldsymbol{\omega} - \boldsymbol{b}_w) \times \boldsymbol{\alpha} - \boldsymbol{b}_w \\ \boldsymbol{v} \\ \boldsymbol{R}_b^e(\hat{\boldsymbol{q}})(\boldsymbol{I}_3 + \boldsymbol{\alpha} \times) \boldsymbol{a}_b + \boldsymbol{g}_e \\ \mathbf{0} \end{bmatrix}. \quad (3.11)$$

The measurement equation reads as

$$\mathbf{y} = \begin{bmatrix} \mathbf{m}_b \\ h_{baro} \\ \mathbf{p}_{GPS} \\ \mathbf{p}_{line} \\ \mathbf{v}_{GPS} \end{bmatrix} = \begin{bmatrix} (\mathbf{I}_3 - \boldsymbol{\alpha}_\times) \mathbf{R}_e^b(\hat{\mathbf{q}}) \mathbf{m}_e \\ p_3 \\ \mathbf{p} \\ \mathbf{p} \\ \mathbf{v} \end{bmatrix}. \quad (3.12)$$

As visible in the measurement equation, multiple sensors measure the position of the kite. This results in estimating the mean of all of this sensors position with the weighting of the covariance matrix entries (see table 3.1). The results of the simulation can be seen in figure 3.8, 3.9 and 3.10. The filter is much better than the AHRS filters because the translational acceleration is now respected. The gyroscope bias is estimated very well, seen in figure 3.9.

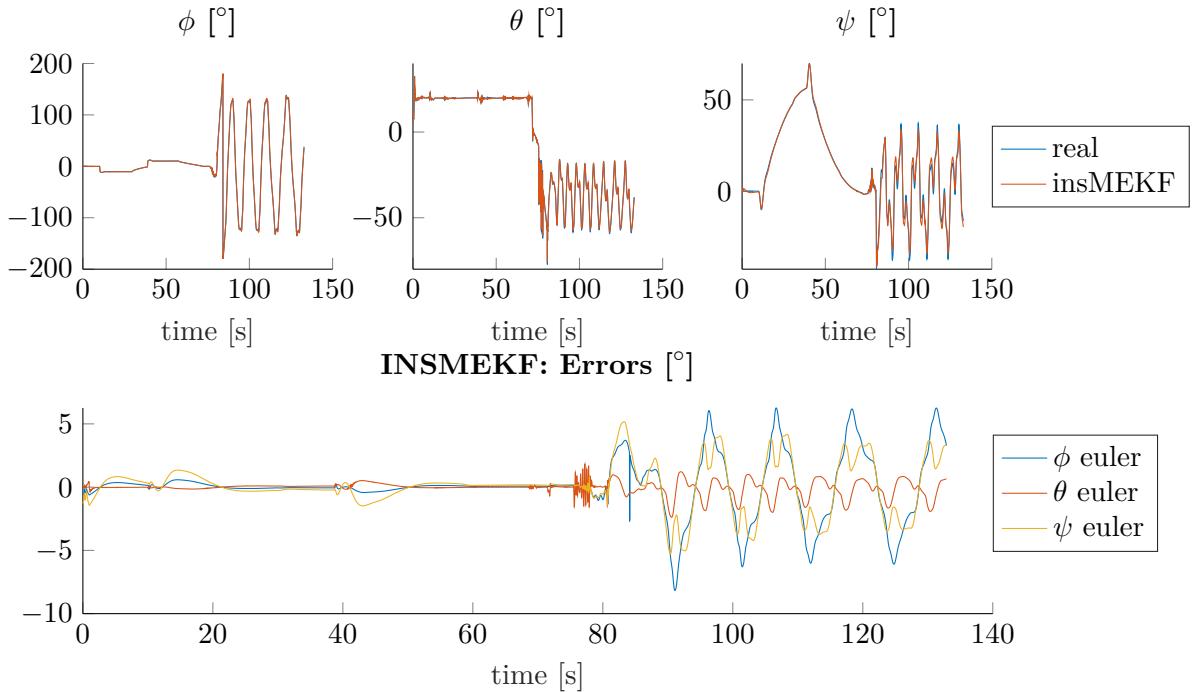


Figure 3.8: Estimated attitude.

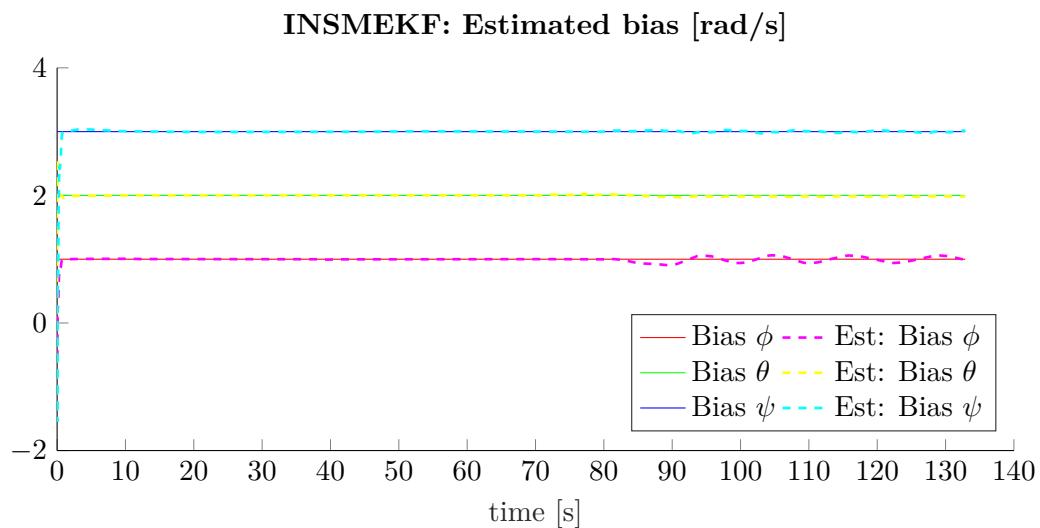


Figure 3.9: Estimated gyroscope bias.

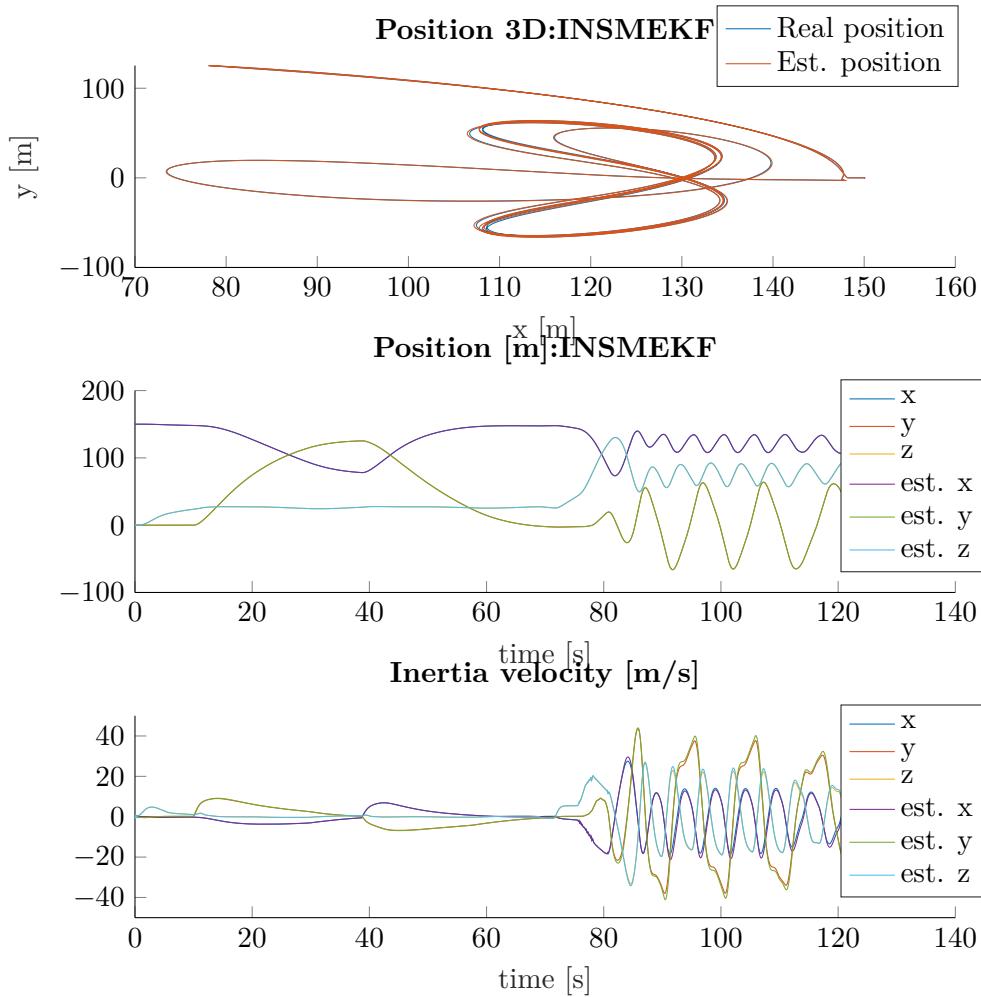


Figure 3.10: Estimated position and velocity.

3.3 Additional limitations

In this section additional limitations are discussed and implemented into the filter to test the performance.

3.3.1 Loop time shorter than sensor reading

Some sensors do not have such high sample rate than the filter. For this case the simulation was extended to use the sample rates in table 3.4. The sample rate of the filter is the same as the gyroscope one. Three different concepts for handling different sample times are evolved.

Sensor / Filter	Sample Rate
Filter loop time	100 Hz
Gyroscope	100 Hz
Accelerometer	100 Hz
Magnetometer	100 Hz
GPS	10 Hz
Barometer	100 Hz
Line angle sensor	100 Hz

Table 3.4: Sample rates of different sensors.

The first filter uses the previous value when no new measurement is available, the second

filter skips the correction step of the not available filter and the third filter integrates the position from the velocity, and uses the previous values for magnetometer, accelerometer and gyroscope. The results of the simulations are visible in table 3.5, 3.6 and 3.7. Seeing in this tables, the attitude estimations from INSMEKF DST and INSMEKF Interpolate are better than ignoring the availability and using old values. Interpolating the position or the velocity in high dynamic trajectories is not a good idea. Better it's to skip the correction step of the GPS module if no measurements are available and just using the tether angle sensor to estimate the position.

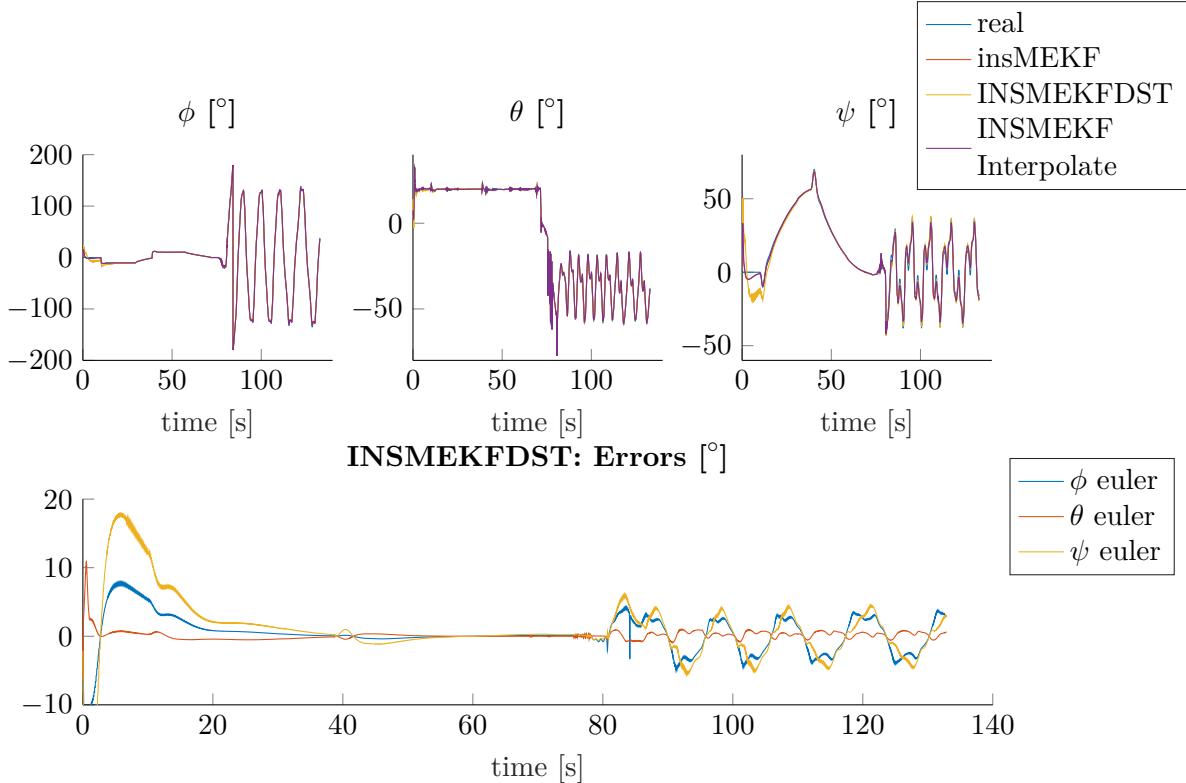


Figure 3.11: Attitude Estimation with different sensor availability handling.

Filter	Name in 3.11	Angle [°]		
		rmsd(ϕ)	rmsd(θ)	rmsd(ψ)
INS MEKF without interpolation	INSMEKF	2.9236	1.2891	6.2544
INS MEKF no measurement step if no measurement available	INSMEKF DST	2.7114	0.90981	5.619
INS MEKF with interpolation	INSMEKF Interpolate	2.7045	0.68226	4.291

Table 3.5: Estimated angle error with different sensor availability handling.

The interpolation of the position is done with the formula

$$p = p_x + v_{GPS} \cdot sampleTime \quad (3.13)$$

where x is a placeholder for GPS, lineAngle or barometer.

Filter	Name in 3.11	Position [m]			
		rmsd(x)	rmsd(y)	rmsd(z)	rmsd(.)
INS MEKF without interpolation	INSMEKF	1.5142	1.4959	0.22702	2.1405
INS MEKF no measurement step if no measurement available	INSMEKFDST	0.77056	0.74912	0.22176	1.0973
INS MEKF with interpolation	INSMEKF Interpolate	1.8183	1.3863	0.33992	2.3116

Table 3.6: Estimated position error with different sensor availability handling.

Filter	Name in 3.11	Velocity [m s^{-1}]			
		rmsd(x)	rmsd(y)	rmsd(z)	rmsd(.)
INS MEKF without interpolation	INSMEKF	0.68635	0.76425	0.37651	1.094
INS MEKF no measurement step if no measurement available	INSMEKFDST	0.75226	0.77643	0.43567	1.1656
INS MEKF with interpolation	INSMEKF Interpolate	1.5774	1.347	0.82233	2.2313

Table 3.7: Estimated velocity error with different sensor availability handling.

3.3.2 GPS module acceleration limitation

Cheap GPS modules are limited to certain parameters like height, velocity and acceleration. As example, the u-blox NEO-6 module is limited to an acceleration of 4 g[44]. The dynamics of the kite is high enough to exceed this limit and so no GPS data are anymore available. This limitation was simulated and can be seen in figure 3.12, where the GPS position is constant for the time the 4 g acceleration is exceeded. The position is constant, because the previous value is used for the filter. The time range between two GPS points is about 1.2 s. Seeing in figure 3.13 or in table 3.8, 3.9 and 3.10 it is visible that the estimation errors for the filter which uses the previous values and the filter which interpolates the position increased a lot compared to the results from section 3.3.1. The filter which skips the correction step do not increase much. This is because if no GPS is available, the tether angle sensor is used. To interpolate the missing values decreases the error, but skipping the measurements which are not available gives much better results.

Filter	Name in 3.14	Angle [$^\circ$]		
		rmsd(ϕ)	rmsd(θ)	rmsd(ψ)
INS MEKF without interpolation	INSMEKF	8.8124	3.23	12.8415
INS MEKF no measurement step if no measurement available	INSMEKFDST	2.4306	0.65178	4.3313
INS MEKF with interpolation	INSMEKF Interpolate	3.5988	1.0983	4.8015

Table 3.8: Estimated angle error with 4 g limitation of the GPS module.

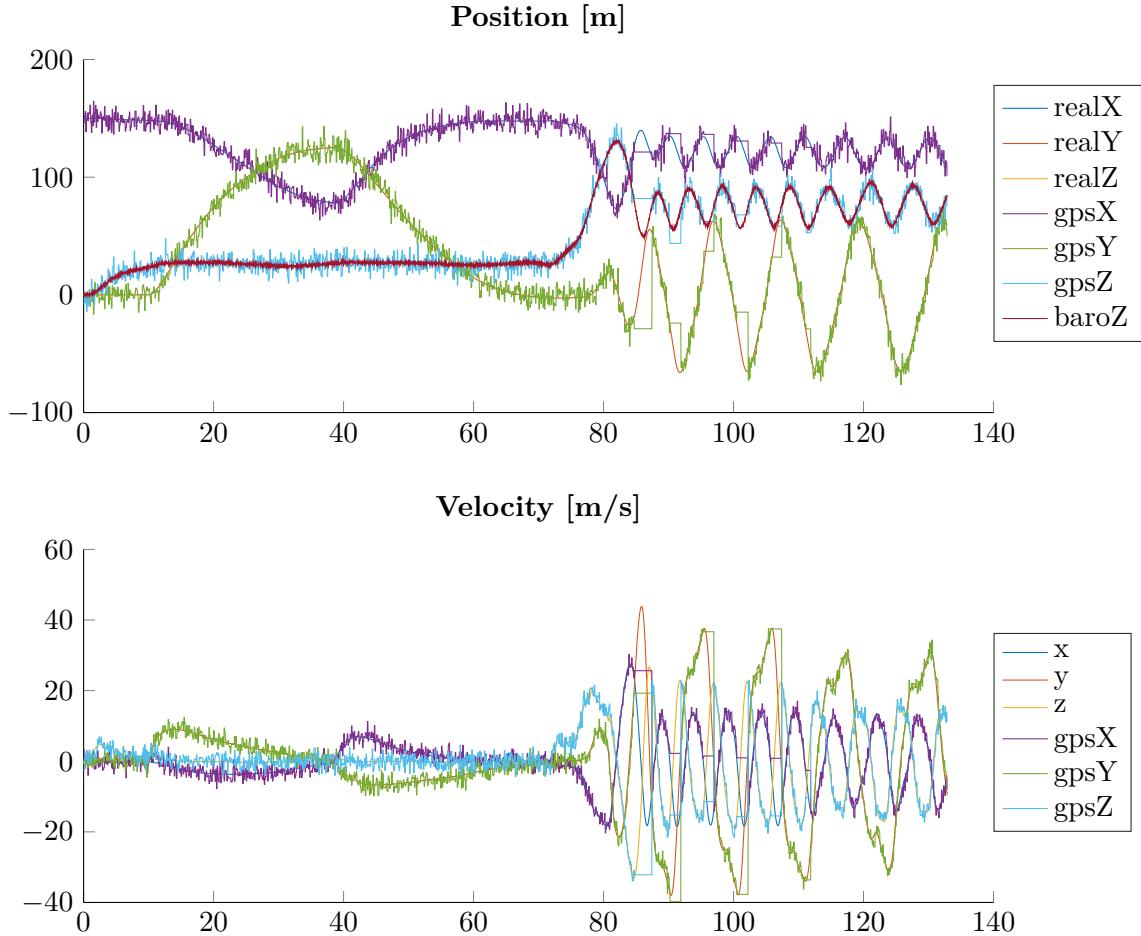


Figure 3.12: GPS Position with limitation to 4 g.

Filter	Name in 3.14	Position [m]			
		rmsd(x)	rmsd(y)	rmsd(z)	rmsd(.)
INS MEKF without interpolation	INSMEKF	3.176	5.4799	0.29372	6.3406
INS MEKF no measurement step if no measurement available	INSMEKFDST	0.84375	0.86099	0.22689	1.2267
INS MEKF with interpolation	INSMEKF Interpolate	4.3232	3.159	0.49217	5.377

Table 3.9: Estimated position error with 4 g limitation of the GPS module.

Filter	Name in 3.14	Velocity [m s^{-1}]			
		rmsd(x)	rmsd(y)	rmsd(z)	rmsd(.)
INS MEKF without interpolation	INSMEKF	7.0452	5.1703	5.0403	10.0882
INS MEKF no measurement step if no measurement available	INSMEKFDST	0.95864	1.0153	0.50761	1.4858
INS MEKF with interpolation	INSMEKF Interpolate	2.1994	2.6406	1.088	3.6047

Table 3.10: Estimated velocity error with 4 g limitation of the GPS module.

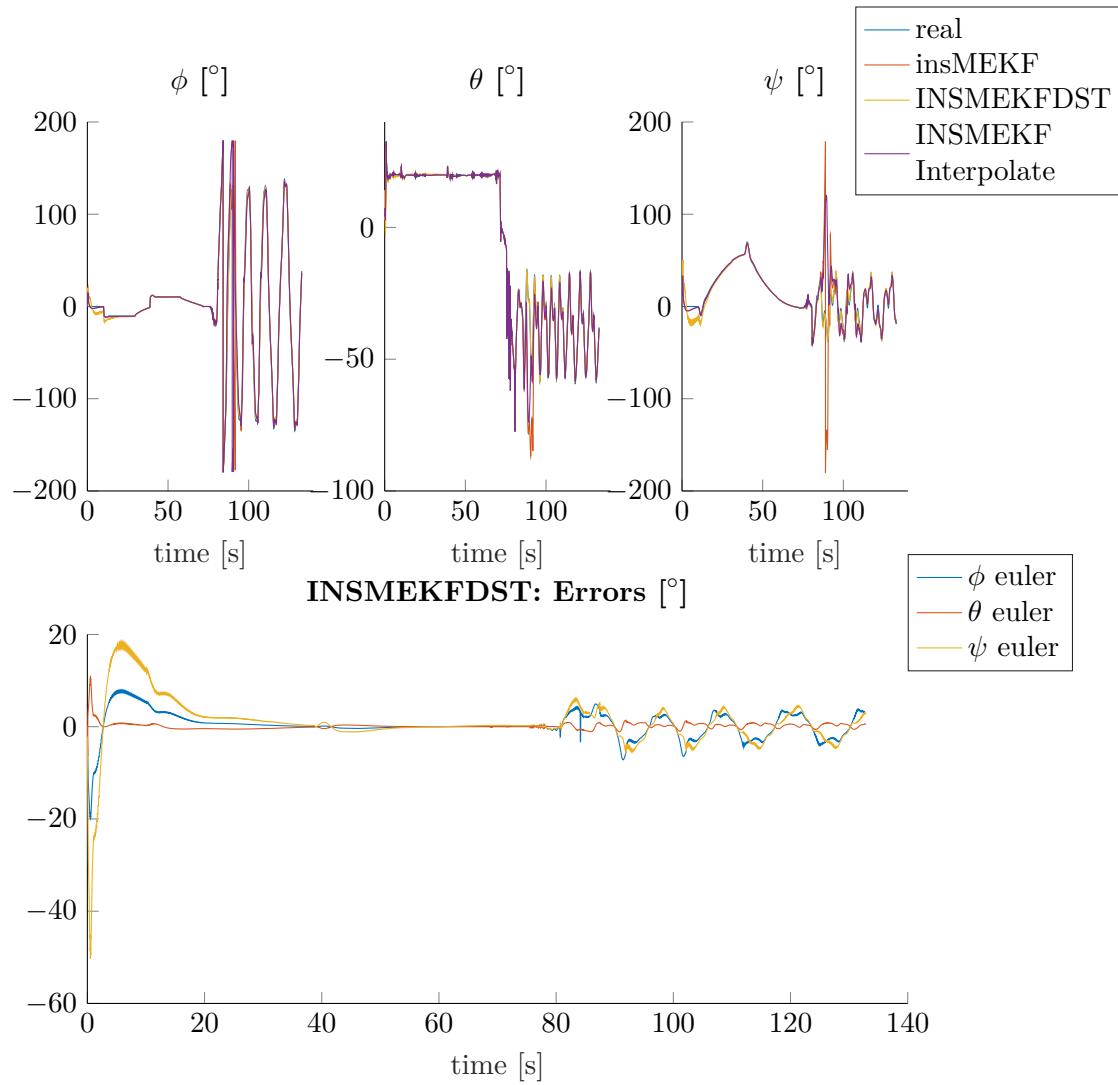


Figure 3.13: Estimated angles of different filters and the error for the filter which does not calculate the correction step, with the GPS module limited to an acceleration of 4g and sample rates defined in section 3.3.1.

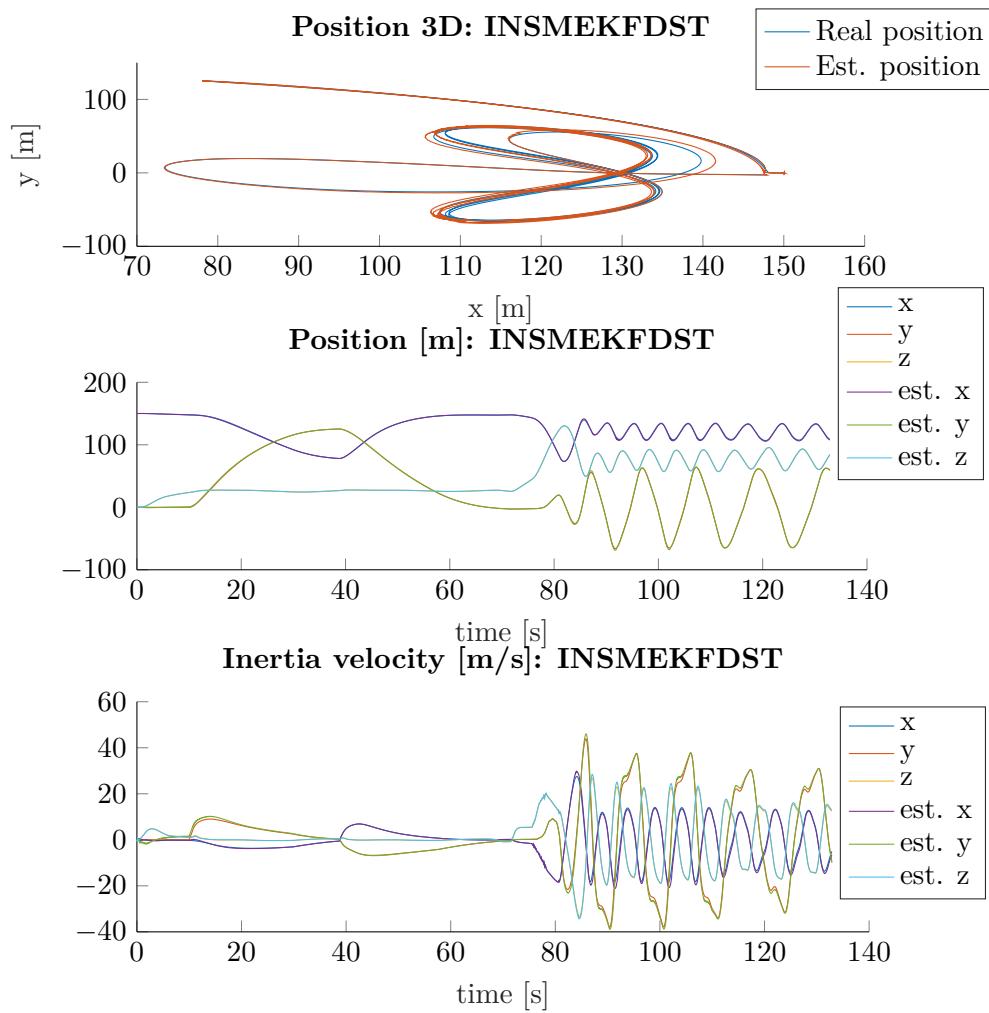


Figure 3.14: Position of INSMEKF DST, with the GPS module limited to an acceleration of $4g$.

3.3.3 Tether sag

In the previous sections it was assumed, the tether is a straight line and the position could be calculated with the equation (2.56) from section 2.2.6. Due to the mass of the tether it's impossible to have a straight line and the tether has a sag. To determine the position of the kite including the compensation of the tether sag, the tether is described as a Catenary curve [45]. This type of curve was developed to describe cables hanging under their own weight[45]. This derivation is first done in 2D which can be seen in figure 3.15 and will be extended to 3D in a second step. The Catenary curve is described in general as

$$y = C2 + a \cosh\left(\frac{1}{a}x + C1\right). \quad (3.14)$$

There are three parameters a , $C1$ and $C2$ which must be determined to estimate the position of the kite. $C1$ and $C2$ are constants of integration and can be found by using defined boundaries. In general the derivative of the Catenary curve is

$$\frac{dy}{dx} = \sinh\left(\frac{1}{a}x + C1\right). \quad (3.15)$$

The line angle sensor measures the tether angle and therefore the gradient of the tether is

$$\tan(\theta_{base}) = \frac{y}{x}. \quad (3.16)$$

$C1$ can be found by using the line angle sensor at the base station (3.16) and the derivative of the Catenary curve (3.15) with $x = 0$ as

$$C1 = \operatorname{asinh}(\tan(\theta_{base})). \quad (3.17)$$

The second integration constant can be calculated with the fact that the tether is connected to the base, which means at position $x = 0$, $y = 0$ holds. Plugging these values into the Catenary equation (3.14) $C2$ is expressed in terms of $C1$ as

$$C2 = -a \cosh(C1). \quad (3.18)$$

The constant a can be calculated in dependence of the kite horizontal position x_{kite} from the second tether angle sensor on the kite as

$$a = \frac{x_{kite}}{\operatorname{asinh}(\tan(\theta_{kite})) - C1}. \quad (3.19)$$

As the length of the tether is known, it's possible to use an algebraic equation of the curve length to determine the horizontal distance of the kite. In general a curve length can be determined with the formula from [46] as

$$l = \int_0^{x_{kite}} \sqrt{1 + f'(x)^2} dx. \quad (3.20)$$

Using this formula and plugging the definitions of $C1$, $C2$ and a into the result we get the following expression for x_{kite}

$$x_{kite} = \frac{2l e^{\operatorname{asinh}(\tan(\theta_{kite}))+C1} \operatorname{asinh}(\tan(\theta_{kite})) - 2C1 l e^{\operatorname{asinh}(\tan(\theta_{kite}))+C1}}{e^{2\operatorname{asinh}(\tan(\theta_{kite}))+C1} - e^{\operatorname{asinh}(\tan(\theta_{kite}))+2C1} + e^{\operatorname{asinh}(\tan(\theta_{kite}))} - e^{C1}}. \quad (3.21)$$

The y value is then calculated with the Catenary curve.

Extending to 3D The Catenary curve is a two dimensional curve, but the position of the kite is in three-dimensional space. For this reason, the x and y positions are combined to get a horizontal distance as

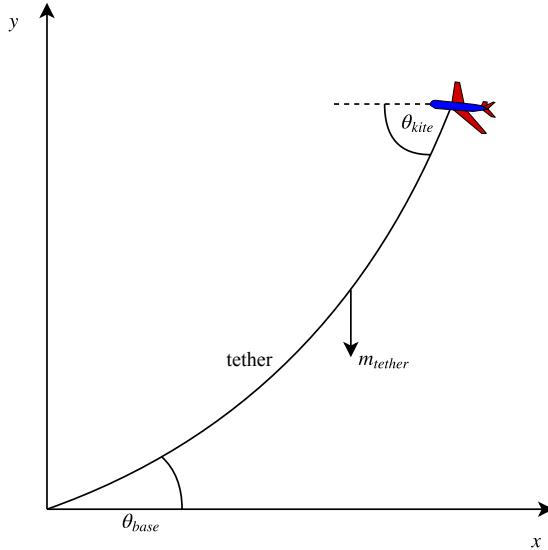
$$x_{hor} = \sqrt{x^2 + y^2}, \quad (3.22)$$

the z component is the y component in the two-dimensional case. x_{hor} can be calculated with formula (3.21). z can then be determined with the catenary equation (3.14). The x and y component can be calculated with the help of the azimuth angle ϕ_{base} , measured from the base as

$$x = x_{hor} \cos(\phi_{base}) \quad (3.23)$$

and

$$y = x_{hor} \sin(\phi_{base}). \quad (3.24)$$



Source: Adapted from [9]

Figure 3.15: Tether sagging due to the mass of the tether.

Simulating tether sag Because no tether angle values in the received simulation data are available, the angles must be simulated. For this the previous steps must be performed in reverse order. Therefore, we introduce a new variable to increase the calculation speed by not calculating every time the sinus hyperbolicus

$$b = \operatorname{asinh}(\tan(\theta_{kite})). \quad (3.25)$$

To map the three-dimensional position to a two-dimensional system we use the horizontal position x_{hor} . With the new variable b and the horizontal position x_{hor} we get a as

$$a = \frac{x_{hor}}{b - C1} \quad (3.26)$$

Putting this equation in the Catenary curve and setting the z component of the position as y we get

$$-z + \frac{\cosh(b)x_{hor}}{b - C1} - \frac{\cosh(C1)x_{hor}}{b - C1} = 0 \quad (3.27)$$

with the unknowns b and $C1$. To get a solution for $C1$ and b we use the curve length equation to get a second equation

$$l - \frac{e^{2b+C1}x1 + (1 - e^{2C1})e^bx_{hor} - e^{C1}x_{hor}}{2e^{b+C1}b - 2C1e^{b+C1}} = 0. \quad (3.28)$$

There is no algebraic solution of this nonlinear equation system, for what reason, this system must be solved numerically. After solving this system, θ_{kite} and θ_{base} can be calculated with equation (3.25) and (3.17). The azimuth angle of the base tether angle sensor can be calculated as

$$\phi_{base} = \tan\left(\frac{y}{x}\right). \quad (3.29)$$

Simulation results

For the simulation the following assumption is used

Assumption 9. *The tether lenght is 155 m at a direct distance from the base station to the kite of 150 m*

The results of the above compensation can be seen in table 3.11, 3.12 and 3.13. The attitude is not falsified much from the sag, but the errors of the x and y position of the kite are falsified, seeing also in figure 3.16. This is, because the position of the line angle sensor is estimated wrong and has a much higher sample frequency as the GPS module, whereas, the position is next to the estimated position from the line angle sensor. The z component is not much falsified, because the barometer has a lower variance than the GPS module and the tether angle sensor. The estimation is near to the estimation of the barometer. The velocity is not falsified, because this state is only compared to the GPS velocity.

Filter	Angle [°]		
	rmsd(ϕ)	rmsd(θ)	rmsd(ψ)
INS MEKF without tether sag compensation	5.3193	3.4124	14.2155
INS MEKF with tether sag compensation	3.1019	1.1386	6.5441

Table 3.11: Estimated angle error with and without tether sag compensation.

Filter	Position [m]			
	rmsd(x)	rmsd(y)	rmsd(z)	rmsd($\ \cdot\ $)
INS MEKF without tether sag compensation	124.2974	60.4464	0.8792	138.2186
INS MEKF with tether sag compensation	1.5188	1.188	0.21792	1.9405

Table 3.12: Estimated position error with and without tether sag compensation.

Filter	Velocity [m s^{-1}]			
	rmsd(x)	rmsd(y)	rmsd(z)	rmsd($\ \cdot\ $)
INS MEKF without tether sag compensation	1.2561	3.2219	0.60557	3.5108
INS MEKF with tether sag compensation	0.93113	1.0264	0.45668	1.4591

Table 3.13: Estimated velocity error with and without tether sag compensation.

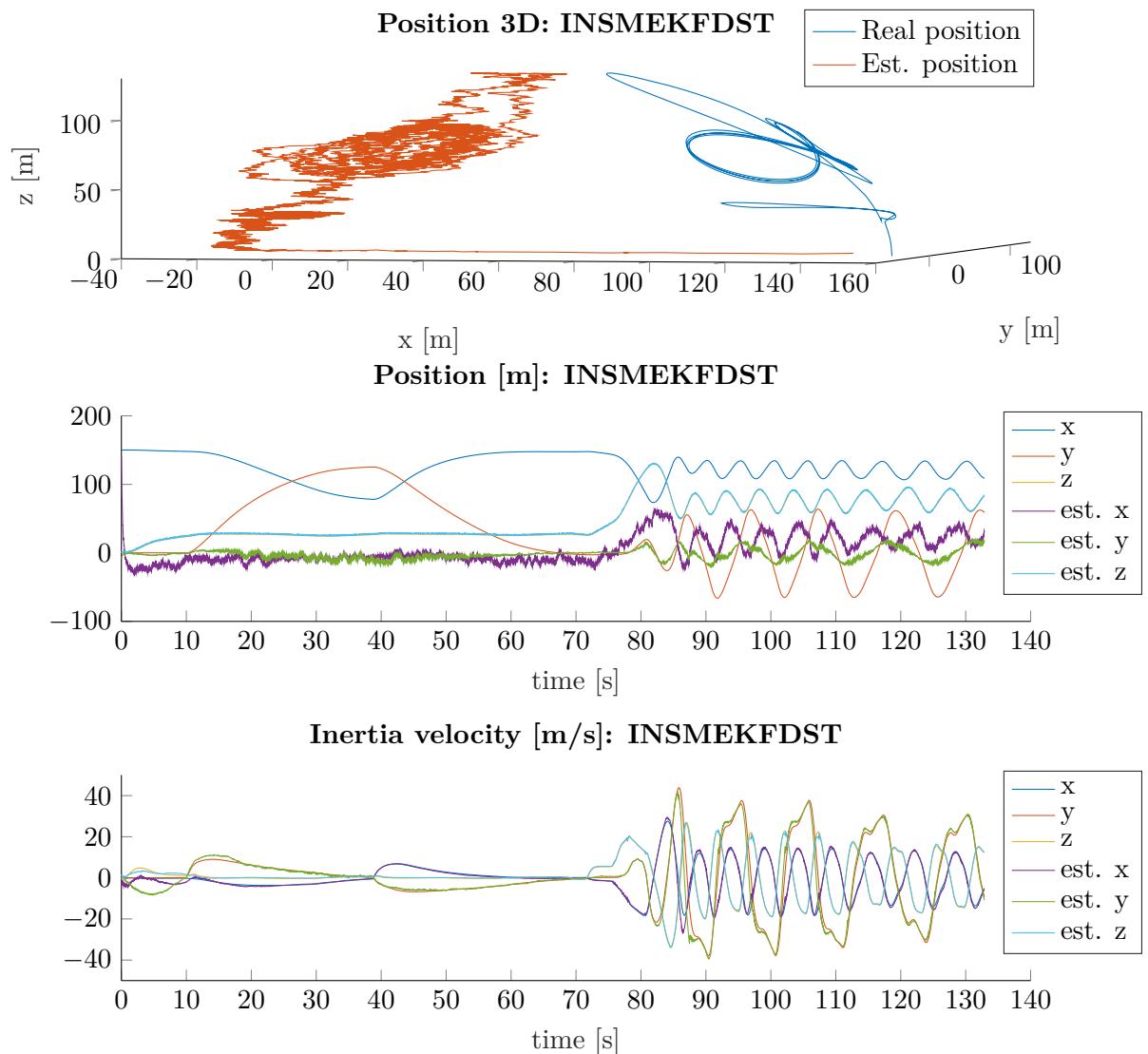


Figure 3.16: Position and velocity of the INS MEKF filter without tether sag compensation.

Chapter 4

Preparation for practical use

4.1 Magnetometer calibration

In this section the magnetometer STMicroelectronics LSM303 is calibrated with the theory from section 2.2.3. To calibrate the magnetometer, measurements must be available, where the sensor was rotated about all axes. The result of this measurement can be seen in figure 4.1 drawn as blue dots (\bullet). The measurement was made outside and away from buildings. So the magnetometer is not much distorted by soft-iron effects, and therefore the measurements create almost a sphere and not an ellipsoid. It's visible, that there is an offset from the origin, the hard iron distortion. After using the equations from section 2.2.3, the magnetometer values create a sphere around the origin, which is shown in figure 4.1 as orange dots (\bullet). Figure 4.2 shows the mounting of an Arduino with the sensor to measure the accuracy of the magnetometer calibration. The “real” angle was measured with a potentiometer which is mounted between the upper and the lower part of the mount. Figure 4.3 and table 4.1 show the results of the calibration. It can be seen, that the hard iron calibration increases the accuracy a lot, but the soft iron calibration has increased the accuracy only a bit, which is because the measurement was made outside and far away from any ferromagnetic materials.

Compensation type	RMSD
Hard iron compensation (—)	6.6495°
Hard and soft iron compensation (—)	5.6996°
Without compensation (—)	11.4674°

Table 4.1: Test of the calibration of the magnetometer.

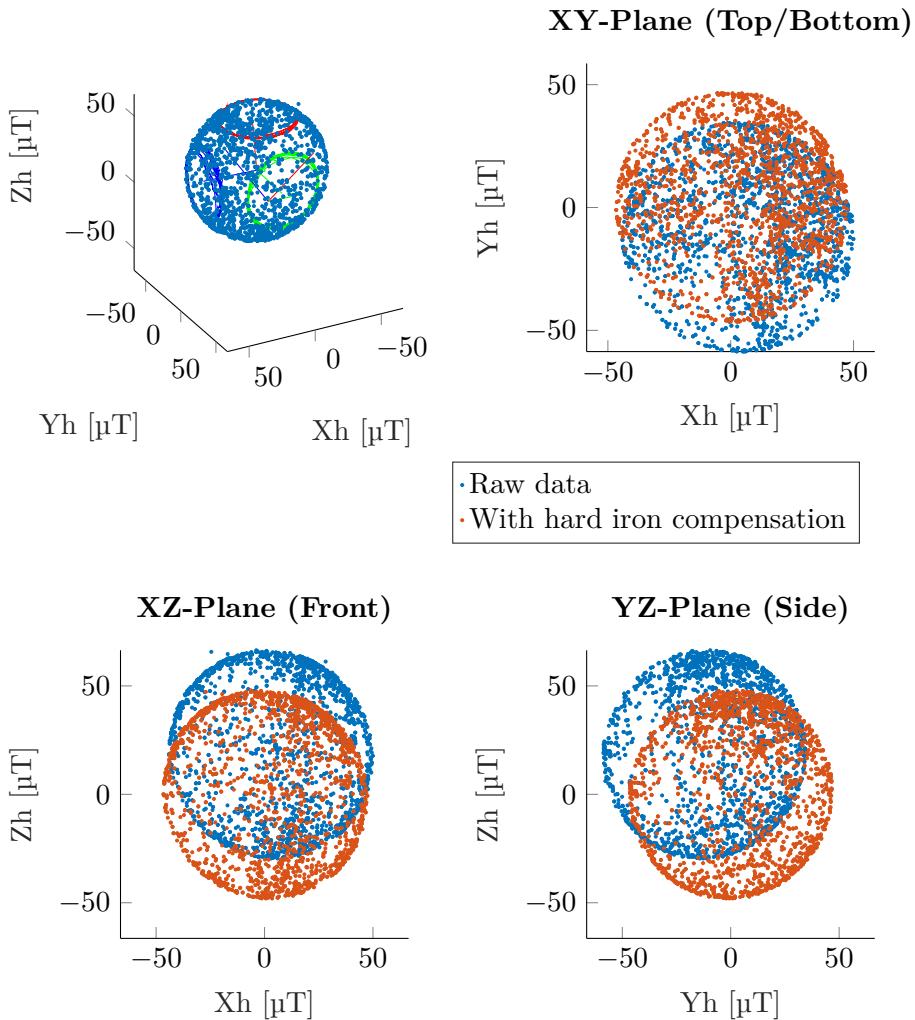


Figure 4.1: Compensating hard iron distortion. The blue dots show the raw magnetometer data, the orange dots show the hard iron calibrated magnetometer data.



Figure 4.2: Testmount to measure the accuracy of the magnetometer calibration.

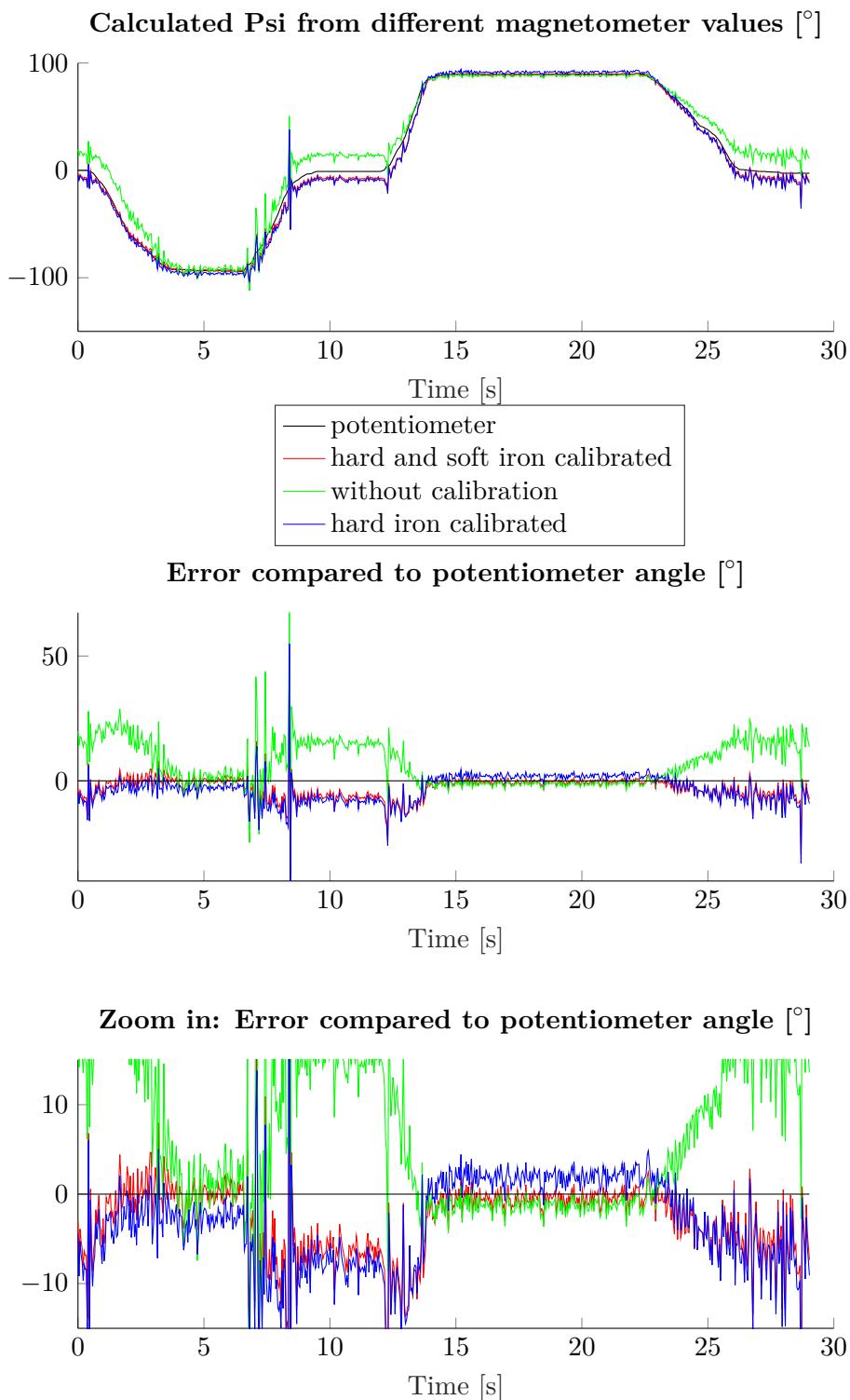


Figure 4.3: Test of the calibration of the magnetometer. The black line displays the reference angle measured with the setup in figure 4.2. The green line shows the uncalibrated angle, the blue line the hard-iron calibrated angle and the red line the hard- and soft-iron calibrated angle.

4.2 C-Code generation

To use the filter code on a microcontroller without matlab, the matlab code must be transferred to a programming language which can be compiled and used on the microcontroller. In the most cases C or C++ is chosen. Matlab has its own C/C++ Code generator, so it is possible to export the code directly. In many cases, model based design with Simulink is made. Therefore, a Simulink model of the Matlab code was created. For this case Simulink offers a block called “MATLABSystem”, which imports a Matlab class. The Matlab class must inherit from the matlab.System class and small changes to the code must be made to be able to use the filter. A second benefit of the Simulink model is, that also “continuous” system models can be simulated. Every model which should be converted to C-Code, must be in a subblock with an “enable” block to discretize the model. An example of a filter in Simulink is shown in figure 4.4

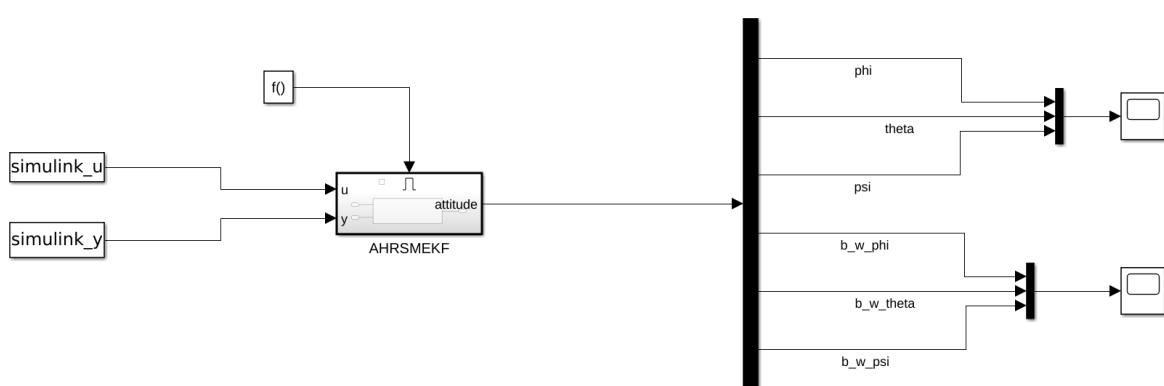


Figure 4.4: Matlab filter simulated in Simulink with the AHRS MEKF as example.

Chapter 5

Conclusion and Outlook

In this thesis AHRS, INS filters in Kalman filter configuration or in complementary filter configuration were modelled and compared. The high translational acceleration in the trajectory of the kite, makes the AHRS filters not suitable because of the high errors between the real states and the estimated states. INS filters include the translational acceleration into the state equations for what reason the estimation errors decrease. Due to the limited sample rate of the sensors, a MEKF was designed to skip the Kalman filters correction step for the sensor which is not available. This decreases the error between the estimated and the real values more than interpolating the missing sensor ones. While the kite describes the desired trajectory, it exceeds the maximum acceleration of the GPS module (4 g) several times per round. In this time, the GPS module isn't anymore available to estimate the position and the velocity. In this case, only the line angle sensor is used to estimate the position. In addition, the tether is modeled as a catenary curve. The MEKF with implemented compensation is compared to the one assuming the tether as a straight line. In this last case, the attitude and the position would present high errors. To get a better model of the tether, including also drag, it could be further designed as a dynamic model, which has already been done by Williams et al. in [47].

Bibliography

- [1] Richard Craig. *Renewables exceed 40% in Germany for 2018*. 2019. URL: <https://www.windpowermonthly.com/article/1522063/renewables-exceed-40-germany-2018> (visited on 04/08/2019).
- [2] Umweltbundesamt. *Erneuerbare Energien in Zahlen*. 2019. URL: <https://www.umweltbundesamt.de/themen/klima-energie/erneuerbare-energien/erneuerbare-energien-in-zahlen#statusquo> (visited on 04/08/2019).
- [3] Benjamin Prünster. *A Comparative Study between Lift Power Kites and Drag Power Kites*. 2016.
- [4] Florian Bauer. *Kitekraft*. 2019. URL: <http://www.kitekraft.de/> (visited on 04/08/2019).
- [5] F. Bauer et al. „Multicopter With Series Connected Propeller Drives“. In: *IEEE Transactions on Control Systems Technology* 26.2 (Mar. 2018), pp. 563–574. ISSN: 1063-6536. DOI: 10.1109/TCST.2017.2679071.
- [6] Markus Johannes Sedelmaier. „Sensor Fusion for an Airborne Wind Energy System“. MA thesis. Technische Universität München, Sept. 19, 2018.
- [7] AweCrosswind. *Crosswind kite power station with fast motion transfer having two wings offshore*. 2012. URL: https://upload.wikimedia.org/wikipedia/commons/9/91/Crosswind%5C_kite%5C_power%5C_station%5C_with%5C_fast%5C_motion%5C_transfer%5C_having%5C_two%5C_wings%5C_offshore.jpg (visited on 05/16/2019).
- [8] Jeffrey D. Barton. „Fundamentals of Small Unmanned Aircraft Flight“. In: *JOHNS HOPKINS APL TECHNICAL DIGEST, VOLUME 31* (2012).
- [9] Auawise. *Aircraft principal axes*. 2019. URL: https://en.wikipedia.org/wiki/Aircraft%5C_principal%5C_axes%5C#/media/File:Yaw%5C_Axis%5C_Corrected.svg (visited on 05/15/2019).
- [10] James Diebel. „Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors“. In: *Matrix* 58 (Apr. 8, 2019).
- [11] James M. Maley. „Multiplicative Quaternion Extended Kalman Filtering for Nonspinning Guided Projectiles“. In: *ARL-TR-6503* (2013).
- [12] F. Landis Markley. „Attitude Error Representations for Kalman Filtering“. In: *Journal of Guidance, Control and Dynamics, Vol. 26, No. 2* (2003).
- [13] Erwan Salaün Philippe Martin. „Generalized Multiplicative Extended Kalman Filter for Aided Attitude and Heading Reference System“. In: *Navigation, and Control Conference, Aug 2010, Toronto, Canada* (2010). URL: <https://hal-mines-paristech.archives-ouvertes.fr/hal-00555580/document>.
- [14] Rachel Muheim. „The Light-Dependent Magnetic Compass“. In: *Photobiology*. Dec. 2007, pp. 465–478. DOI: 10.1007/978-0-387-72655-7_17. URL: https://www.researchgate.net/publication/226174692_The_Light-Dependent_Magnetic_Compass.

- [15] National Centers for Environmental Information. *Magnetic Field Calculators*. 2019. URL: <https://www.ngdc.noaa.gov/geomag/calculators/magcalc.shtml#declination> (visited on 04/09/2019).
- [16] Magnetic-Declination.com. *Find the magnetic declination at your location*. 2019. URL: <http://www.magnetic-declination.com/> (visited on 04/09/2019).
- [17] RockMagnetist. *World Magnetic Inclination 2010*. 2010. URL: https://de.wikipedia.org/wiki/Datei:World%5C_Magnetic%5C_Inclination%5C_2010.pdf (visited on 05/15/2019).
- [18] Christopher Konvalin. *Compensating for Tilt, Hard Iron and Soft Iron Effects*. Mar. 29, 2019.
- [19] Talat Ozyagcilar. „Calibrating an eCompass in the Presence of Hard- and Soft-Iron Interference“. In: *Freescale Semiconductor Application Note*. Mar. 29, 2019.
- [20] Talat Ozyagcilar. „Implementing a Tilt-Compensated eCompass using Accelerometer and Magnetometer Sensors“. In: *Application Note AN4248*. 2015. URL: https://cache.freescale.com/files/sensors/doc/app_note/AN4248.pdf.
- [21] A. Turner; D et al. *An Algorithm for Fitting an Ellipsoid to Data*. Feb. 1970.
- [22] Li; Qingde and G Griffiths; J. *Least squares ellipsoid specific fitting*. Feb. 2004. DOI: 10.1109/GMAP.2004.1290055.
- [23] Yuri. *Ellipsoid fit*. 2015. URL: <https://de.mathworks.com/matlabcentral/fileexchange/24693-ellipsoid-fit> (visited on 04/09/2019).
- [24] Bot Thoughts. *Magnetometer Soft- and Hard-Iron Calibration*. 2012. URL: <https://diydrones.com/forum/topics/magnetometer-soft-and-hard-iron-calibration?id=705844%3ATopic%3A773589&page=1#comments> (visited on 05/09/2019).
- [25] Michael Schmandt. *GIS Commons - Ch.2: Input*. Ed. by Michael Schmandt. 2019. URL: <https://giscommons.org/chapter-2-input/> (visited on 06/14/2019).
- [26] Bosch Sensortec. *BMP180 Digital pressure sensor*. Tech. rep. Bosch GmbH, May 9, 2019. URL: https://ae-bst.resource.bosch.com/media/_tech/media/datasheets/BST-BMP180-DS000.pdf.
- [27] Andeggs. *Spherical coordinate system*. 2009. URL: https://en.wikipedia.org/wiki/Spherical%5C_coordinate%5C_system%5C#/media/File:3D%5C_Spherical.svg (visited on 05/15/2019).
- [28] R. Mahony, T. Hamel, and J. Pflimlin. „Nonlinear Complementary Filters on the Special Orthogonal Group“. In: *IEEE Transactions on Automatic Control* 53.5 (June 2008), pp. 1203–1218. ISSN: 0018-9286. DOI: 10.1109/TAC.2008.923738.
- [29] Wolfgang Utschick. „Lecture notes in Statistical Signal Processing“. 2017.
- [30] Ian Reid. „Estimation II“. 2001. URL: <http://www.robots.ox.ac.uk/~ian/Teaching/Estimation/LectureNotes2.pdf>.
- [31] Kurt Geelen. „Design and Operation of Airborne Wind Energy Systems“. PhD thesis. KU Leuven – Faculty of Engineering Science, May 16, 2019.
- [32] Robert Grover; Brown Patrick; Y.C.Hwang. *Introduction to Random Signals and Applied Kalman Filtering*. New York, NY: Wiley, 2012.
- [33] Richard M. Murray. *Optimization-Based Control*. Tech. rep. Control and Dynamical Systems California Institute of Technology, May 9, 2019. URL: http://www.cds.caltech.edu/~murray/books/AM08/pdf/obc-kalman_22Dec09.pdf.
- [34] F. Bauer. *Luenberger-Beobachter und Extended Kalman-Filter: Ein Vergleich*. Tech. rep. Technical University of Munich, 2013.

- [35] Siegfried M.Rump. *Inversion of Extremely Ill-Conditioned Matricesin Floating-Point.* 2019. URL: <http://www.ti3.tuhh.de/paper/rump/Ru08a.pdf> (visited on 05/16/2019).
- [36] Gabriel A. Terejanu. *Extended Kalman Filter Tutorial.* 2010. URL: <https://www.cse.sc.edu/~terejanu/files/tutorialEKF.pdf>.
- [37] R. Mahony, T. Hamel, and J. Pflimlin. „Complementary filter design on the special orthogonal group SO(3)“. In: *Proceedings of the 44th IEEE Conference on Decision and Control.* Dec. 2005, pp. 1477–1484. DOI: 10.1109/CDC.2005.1582367.
- [38] PX4DevTeam. *Pixhawk 4.* 2019. URL: https://docs.px4.io/en/flight%5C_controller/pixhawk4.html (visited on 05/14/2019).
- [39] Paul Riseborough. *GenerateNavFilterEquations.* 2019. URL: <https://github.com/PX4/ecl/blob/master/EKF/matlab/scripts/Inertial%20Nav%20EKF/GenerateNavFilterEquation.m> (visited on 05/17/2019).
- [40] Hellozwy. *Pixhawk-optional-peripherals with i-rtk.* 2019. URL: https://upload.wikimedia.org/wikipedia/commons/8/8d/Pixhawk-optional-peripherals%5C_with%5C_i-rtk.jpg (visited on 05/15/2019).
- [41] ArduPilot Dev Team. *Extended Kalman Filter Navigation Overview and Tuning.* 2019. URL: <http://ardupilot.org/dev/docs/extended-kalman-filter.html> (visited on 05/14/2019).
- [42] PX4DevTeam. *PX4 Development Guide.* 2019. URL: https://dev.px4.io/en/tutorials/tuning_the_ecl_ekf.html (visited on 05/17/2019).
- [43] PX4DroneAutopilotContributors. *PX4 Firmware.* 2019. URL: <https://github.com/PX4/Firmware> (visited on 05/10/2019).
- [44] u-blox. *NEO-6.* 2019. URL: [https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_\(GPS_G6-HW-09005\).pdf](https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_(GPS_G6-HW-09005).pdf) (visited on 05/02/2019).
- [45] Glenn Research Center. *Control Line Equations.* Ed. by Nancy Hall. 2015. URL: <https://www.grc.nasa.gov/www/k-12/airplane/kitesag.html> (visited on 04/23/2019).
- [46] Paul Dawkins. *Arc Length.* 2018. URL: <http://tutorial.math.lamar.edu/Classes/CalcII/ArcLength.aspx> (visited on 05/09/2019).
- [47] W. Ockels P. Williams B. Lansdorp. „Modeling and control of a kite on a variable length flexible inelastic tether“. In: *Collection of Technical Papers - 2007 AIAA Modeling and Simulation Technologies Conference 2* (Apr. 15, 2019). DOI: 10.2514/6.2007-6705.