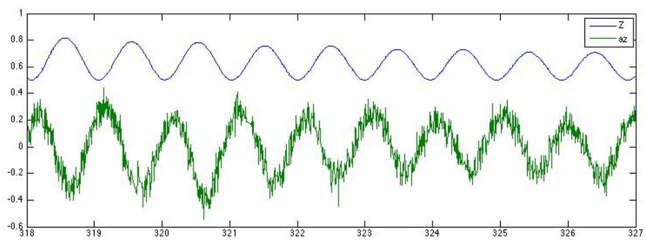
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Automatic Control Laboratory
Prof. Roy Smith

Springterm 2012

Semester Thesis

Sensor Fusion / State Estimation for a Kite Power Plant



Authors:
Roman Patscheider
Fabian Hilti

Advisers:
Aldo Zgraggen
Corey Houle

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für Automatik

NR.1 **YEAR, FS/HS**	**SEMESTERARBEIT**	**IfA-Nr. 20001**

Author	**Roman Patscheider** **Fabian Hilti** Office place	**Name@control.ee.ethz.ch** **Name@control.ee.ethz.ch**
Supervisor	Aldo Zgraggen, Corey Houle	zgraggen@control.ee.ethz.ch, corey.houle@fhnw.ch

Issue Date**:**

Termination Date:

Sensor Fusion / State Estimation for a Kite Power Plant

Description

Working within the context of the SwissKitePower project,

- Develop and implement a data fusion algorithm that provides a real-time estimation of the kite's position, velocity and orientation using the output from multiple sensors.
- Perform laboratory and field tests to validate and quantify the performance boundaries of the estimator.
- Document the results in a report and presentation.

Tasks

Sensor Selection and preliminary testing:

Researchers at FHNW and ETH have already selected and procured a number of sensors and performed preliminary tests. These sensors include:

- Xsens – Commercial GPS + IMU with extended Kalman filter implemented in on-board DSP. http://www.xsens.com/en/general/mti-g

- ArduPilot – Open source GPS + IMU system with DCM (direction cosine matrix) calculation implemented in on-board microprocessor. http://diydrones.com/profiles/blogs/ardupilot-mega-home-page

- X-IMU – Early commercial IMU with integrated storage and Bluetooth, some sensor fusion implemented on-board. http://www.x-io.co.uk/node/9

In addition, the following sensors have been ordered and will be implemented and tested on the FHNW groundstation during the next set of bachelor thesis projects:

- Line angle sensors from TWK - Both the vertical and horizontal angles and angular rates of change of the kite line

Author

Roman Patscheider Name@control.ee.ethz.ch
Fabian Hilti Name@control.ee.ethz.ch

Office place

will be measured using the following sensors. When combined with the length of the line, an estimation of kite position and velocity are possible.
http://www.twk.de/data/pdf/11278fe0.pdf

- Differential GPS system – A DGPS from Novatel has been ordered which consists of two receivers, one which will go on the kite and another on the ground which measures and transmits the correction information.
  http://www.novatel.com/assets/Documents/Papers/OEMStar.pdf

Working with the FHNW students, these systems should be tested and the results analyzed to determine the best suitable combination of sensors for the system. Additional sensors, such as the PixIMU from ETH and the new version of the ArduIMU can also be tested and potentially used in the system. First testing can be made using a centrifuge, which is available at FHNW to understand how the various GPSs and IMU's perform under high g-loads and dynamic conditions. Preliminary tests of this nature have been performed and their results documented in a report which is available.

State Estimator Development:

Based on the results of the first task, a first version of the estimation software should be developed and implemented on an appropriate platform. Most likely this can be done on a PC using Labview or Matlab to acquire and process the incoming data streams and to perform the calculations required. A definition should be made for what sensor values will be passed from the FHNW groundstation to the PC which will perform the calculations as well as what form of output will be given. It is possible that not all sensors are available and the algorithm has to be able to deal with different sensor setups. This should be defined during an initialization phase. Different state estimation algorithms should be implemented and tested. Care should be taken how to evaluate the performance of the different algorithms. As a start the conclusion of the Master Thesis of Héjj Andreás "Kalman-filter based position and attitude estimation algorithms for an Inertial Measurement Unit" can be used. From there on it has to be investigated how we can use the model information of the kite system to improve the state estimation. It can either be used for the state propagation of the INS algorithm directly or apply another Kalman filter in an outer loop to estimate the trajectory.

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für Automatik

NR.1
YEAR, FS/HS

SEMESTERARBEIT

IfA-Nr. 20001

Author **Roman Patscheider** Name@control.ee.ethz.ch
Fabian Hilti Name@control.ee.ethz.ch

Office place

Testing and Validation:

Working with FHNW students and staff, the estimator should be tested in the loop and its output used in some preliminary stabilizing and tracking controllers. The implementation of the controllers themselves will be the responsibility of the project supervisors but the estimator should provide a robust estimate of kite position, velocity and orientation so that the appropriate control actions can be calculated. The performance goals to be achieved are:

- Kite stabilized at zenith for > 1 min.
- Figures of eight flown at constant line length for > 1 min.

Procedures

Time schedule

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für Automatik

NR.1 **YEAR, FS/HS**	# SEMESTERARBEIT	**IfA-Nr. 20001**

Author	**Roman Patscheider** **Fabian Hilti**	Name@control.ee.ethz.ch Name@control.ee.ethz.ch
	Office place	

Oral presentation Date

Signatures

SUPERVISING PROFESSOR Prof. Morari, Institut für Automatik

SUPERVISOR Name

STUDENT Name

Contents

Task i

Abstract viii

1 Introduction 1
1.1 Kite Power in General and the SwissKitePower Project 1
1.2 Related Work 2
1.3 Motivation and Methods 2

2 IMUs 4
2.1 Centrifuge Test 5
2.1.1 Set-up 5
2.1.2 Result 5
2.1.3 Conclusion 8

3 State Estimation 11
3.1 Extended Kalman Filter 11
3.2 Algorithm for State Estimation 15
3.3 State Estimation Model 20
3.4 Sensor Estimation Model 23

4 Results - Vicon Test 25
4.1 Set up 25
4.2 Evaluation 25
4.2.1 Noise of the GPS 26
4.2.2 GPS outage 27
4.2.3 The MTi-G unit versus the PX4 unit 28

5 Conclusion 35

A Data Sheets of the PX4 sensors 38
A.1 Data Sheet of the Accelerometer 38
A.2 Data Sheet of the Gyroscope 40
A.3 Data Sheet of the Magnetometer 42

B Data Sheets of the x-IMU sensors **44**
B.1 Data Sheet of the Gyroscope 44
B.2 Data Sheet of the Accelerometer and Magnetometer 46

List of Figures

2.1 GPS Position . . . 7
2.2 GPS velocity . . . 8
2.3 Accelerometer . . . 9
2.4 Gyrometer . . . 9
2.5 Magnetometer . . . 10
2.6 A closer look in time on the three IMU sensors. . . . 10

3.1 A simple structure of the algorithm . . . 12
3.2 The prediction and the correction step with their corresponding equations. . . . 13
3.3 The prediction and correction step of the EKF with their corresponding equations. . . . 14
3.4 The definition of the inertial and body frame as well as the convention for the angles in the pendulum model. . . . 16
3.5 The graphical represenetation of the algorithm's structure. . . . 19
3.6 The black dot represents the sensor, while r is the distance between the center of mass of the pendulum and the sensor. . . . 24

4.1 Pendulum set up. . . . 26
4.2 Box with sensors. . . . 27
4.3 GPS signal with a SNR of 5 dB. . . . 28
4.4 GPS signal with a SNR of 25 dB. . . . 29
4.5 Error in position and orientation using a GPS signal with a SNR of 5 dB. . . . 30
4.6 Error in position and orientation using a GPS signal with a SNR of 25 dB. . . . 31
4.7 Estimation of the states without GPS signal. . . . 32
4.8 Estimation of the states without GPS and the help of the PX4 IMU. . . . 33
4.9 Error of the estimation of the states without GPS and the help of the PX4 IMU. . . . 34

Abstract

Bla Bla Bla

Chapter 1

Introduction

1.1 Kite Power in General and the SwissKitePower Project

At a time when windmills were already quite commonly used for power generation, Loyd came up with the idea to use kites to convert wind energy to electricity. In 1980 he wrote a seminal paper exploring the possibility of generating electrical power using the pulling force of tethered airfoils, i.e., kites [1]. He describes his concept as follows:

> A kite's aerodynamic surface converts wind energy into motion of the kite. This motion may be converted into useful power by driving turbines on the kite or by pulling a load on the ground. [...] Not simply facing into the wind, such kites would fly a closed path downwind from the tether point. The kite's motion would be approximately transverse to the wind, in the same sense that a wind turbine's blade moves transverse to the wind. The crosswind airspeed of a kite with this trajectory is increased above the wind speed by the lift-to-drag ratio L/D. The resultant aerodynamic lift is sufficient to support a kite and to generate power. [1]

Today, several research groups around the world are investigating this subject and working on prototypes. For example the University of Torino have already tested a prototype [2] and Massimo Ippolito has founded a company named KiteGen that is also located in Torino. At the University of Delft the group of Prof. Dr. Wubbo Oeckels is developing kites to produce energy with "Laddermills" [3] and at the K. U. Leuven the group of Prof. Moritz Diehl is working on the Highwind project [4]. Furthermore the SkySails company is already using wind power to pull large cargo ships [5] and Ampyx Power, a spin-off from T.U.Delft is working on their PowerPlane device.

In Switzerland the research efforts are coordinated in the SwissKitePower Project [6]. SwissKitePower is a collaborative research and development project between FHNW (University of Applied Sciences Northwestern Switzerland), ETHZ (Swiss

Federal Institute of Technology Zurich), EMPA (Swiss Federal Laboratories for Materials Testing and Research) and Alstom Switzerland AG. The goals of the project are to develop 'novel wind energy extraction technology' using tethered airfoils, or kites, and to promote this innovative new technology to the world.

1.2 Related Work

There are many publications on the topic of IMU/GPS data fusion for navigation purposes. As a starting point of our work we used the Master Thesis of András Héjj "Kalman-filter based position and attitude estimation algorithms for an Inertial Measurement Unit" [7]. In an other Master project at the University of Southern Denmark Ushanthan Jeyabalan implemented a Kalman Filter using a spherical pendulum model [8]. Several other publications investigated the use of a Kalman Filter for IMU/GPS data fusion in different applications. Sukkarieh applied it to land vehicles [9], Kim used it for unmanned areal vehicles in highly dynamic flight situations [10] and Crassidis used a Sigma Point Kalman Filter and compared it to an EKF. This list of course is not exhaustive. A much broader overview over the various approaches to multisensor data fusion in target tracking can be found in the survey article of Smith and Singh [11].

1.3 Motivation and Methods

In order to successfully implement a control algorithm on a kite, it is essential that a precise and fast position estimation is available. Due to the slow update rate of the GPS units and their limited reliability, a Kalman Filter will be used to integrate IMU measurements, which consist of acceleration, rate of turn and magnetic field, with the GPS position and velocity estimations. A standard Kalman Filter, one that could also be used in an airplane for example, assumes no knowledge aboout forces acting on the body. Such an estimator will already enhance the position estimation from the raw GPS input since it incorporates more measurements. However, a kite can only move on a restricted surface due to it being thethered to the groundstation. An extended Kalman Filter could take advantage of that knowledge and further improve the estimator's performance. Further integrating an aerodynamical model of the kite could give more information about the forces acting on it and would reduce the estimator's dependency on precise sensor output.

In order to be able to compare the performance of the different estimators, it is neccessary to know the exact location and orientation of the kite at all times. In practice it is nearly impossible to obtain such a ground truth for a kite flying around in the air. Therefore we decided to investigate the benefits of an accurate physical model on the performance of a Kalman Filter using box suspended by a string. (See figure ...) This setup can be modeled as a spherical pendulum which has the advantage that all the forces acting on it are known. In addition, we are able to test the algorithms indoors using a Vicon System (add reference!!) that gives us a very accurate ground truth of the box's position and orientation. Since there is obviously no GPS reception indoors, the GPS output needs to be simulated using the Vicon's position output, reducing its frequency and adding an appropriate level of noise.

Chapter 2

IMUs

For estimating the kite's state measurements from different sensors are needed. In so called inertial measurements units (IMU) several sensors are embedded. The most important difference between the the IMUs are in what sensors are embedded, with which rate do they provide the data, what is their range and sensitivity and how much signal processing is already done by the IMU.

The Swiss Kite Power Project has the following options of IMUs to choose:

MTi-G The MTi-G development kit is a commercial product from the Dutch company Xsens. The unit has an accelerometer, gyroscope, magnetometer, pressure sensor and GPS with antenna included. It exist two output data formats. It can be chosen wether the output is raw data or calibrated data. The calibrated data from the sensors without GPS gives the data in $[m/s^2]$ and takes a offset calibration test into account. In addition to that is the GPS data processed with an Extended Kalman Filter in the calibrated output mode.

PX4 The PX4 is a open-source/open-hardware IMU used and developed by the PIXHAWK Project of the Computer Vision and Geometry Lab of ETH Zurich. The unit contains a temperature and pressure sensor, an accelerometer, a gyroscope and a magnetometer. The IMU additionally provides a counter for each sensor output. The output of the sensors is raw data. In the estimation algorithm (see chapter 3) only the embedded accelerometer a BMA180 from Bosch (see the data sheet in appendix A.1), the gyroscope a L3GD20 from STMicroelectronics (see the data sheet in appendix A.2) and the magnetometer a HMC5883L from Honeywell(see the data sheet in appendix A.3) is used.

x-IMU The x-IMU is a product of the company x-io Technologies. It has as well a temperature sensor, a accelerometer, a gyroscope and a magnetometer. The setting of the x-IMU was done by the FHNW. It provides only raw data. The data sheets of the sensors can be found in the appendix B.

2.1 Centrifuge Test

There are several performance requirements on an IMU installed on a kite. One of them is the dependency on g-loads because on a kite several g loads can appear. Therefore a centrifuge test was carried out in corporation with the Fachhochschule Nordwestschweiz (FHNW)

2.1.1 Set-up

The setting of the centrifuge is described in (âĂę..Figure.....). It is provided by the FHNW. A arm is rotated by a motor. The IMUs are set into a box at the end of the arm. The sensors are put next to each other to have the almost the same measurements in all sensors. The Box can be seen on the picture in (âĂę.Figure....). Obviously is the motion of the box describing a circle with a radius (he distance between the center and the center of the box) of $0.75m$. The motor is able to rotate the arm with a velocity of about $10m/s$. With formula $a = \omega * r^2$ we get maximum acceleration of about $8g$. There is no software for the data collection of the measured velocity implemented. Therefore we have no ground truth to compare the sensor data with. The motor is driven in order to have 5 steps between $1.62g$ and $7.8g$ $(ca.1.6g; 2.4g3.7g; 5.4g; 6.9g; 7.8g)$ of centripetal acceleration.

The PX4 and Xsens are connected together. Both, the PX4 and the Xsens measurements are written on the SD card which is attached on the PX4 Unit. The timestamp is taken from the GPS clock for the Xsens as well as for the PX4 sensor data at the time when they are written on the SD card. This results in easy and accurate synchronization between the two IMUs. To get the time synchronized with the third IMU, the x-IMU is synchronized to the computer's time. Additionally the 3 sensors are hit for having a estimation of how accurate the synchronization is.

2.1.2 Result

The raw data from the x-IMU is scaled by Raphael Mueller bringing it in the common units m/s^2 for the accelerometer, deg/s for the gyroscope and gauss for the magnetometer.

The PX4 was set by the group of the PixHawk Project. How the output of the sensors have to be scaled to bring them in the required units is shown in table 4.1. The scale factor describes by how much the output has to be multiplied to get the data converted in the units described in 4th row. The MTi-G has also raw output data but no clear explanation in the data sheet (âĂę..) on how to get to the required units. Therefore, were the MTi-G data scaled to bring it more or less to the same order as the other two IMUs are. In (...figure...) we can see the plot of the GPS data. Until 4.83e4 s is the centrifuge not in motion. The mean value of this period is the taken as the ground truth to calculate the the average error of the GPS signal. The mean value of the north component is $47.48deg$ for the east component $8.2138deg$ and for the down component $415.74m$. The coordinates of Winidsch, where the

Sensor	Sensitivity	Scale Factor	Unit	Comment
Accelerometer BMA180	$2048LSB/g$	$output * 9.81/2048$	m/s^2	Sensitivity range: $+/-4g$
Gyroscope L3GD20	$17.5mdps/digit$	$output * 17.5/1000$	deg/s	
Magnetometer HMC5883L	$1370LSB/g$	$output/1370$	gauss	

Table 2.1: This table shows how the raw data output of the PX4 has to be sclaed to get the right units.

test takes place is $47.4806deg$, $8.2222deg$ and $357m$. By taking the average absolute difference between the GPS data and the mean value calculated before, an error of the GPS can be estimated: $1.08 * 10^{-5}deg$, $1.37 * 10^{-5}deg$ and $2.49m$ for north, east, down respectively. After $4.83 * 10^4 s$ the noise is increasing. At this time the centrifuge starts rotating. The calculated errors in this period are: $1.68 * 10^{-5}deg$, $2.27 * 10^{-5}deg$ and $2.89m$ for north east, down respectively. After $4.87 * 10^4 s$ the noise look very high. This is when we have a centripetal acceleration of more than $6.7g$. The estimated noise is: $1.04 * 10^{-4}deg$, $1.02 * 10^{-4}deg$ and $13.01m$. Since in non of this periods a cosine of sine behavior of the north and east component can be seen all three components should be stable in the ideal case. The error values a summarized in table 2.2. Due to the fact that the velocity is the derivative of

Period	Average error (north/east/down)
$4.80 * 10^4 s - 4.83 * 10^4 s$	$1.08 * 10^{-5}deg/1.37 * 10^{-5}deg/2.49m$
$4.83 * 10^4 s 4.87 * 10^4 s$	$1.68 * 10^{-5}deg/2.27 * 10^{-5}deg/2.89m$
$4.87 * 10^4 s 4.91 * 10^4 s$	$1.04 * 10^{-4}deg/1.02 * 10^{-4}deg/13.01m$

Table 2.2: The average error of the postion in different time segments

the position of the GPS the amplitude of the error behaves similar. The periods with the correspondent average error is shown in table 2.3. In figure 2.3 and figure

Period	Average error(x/y/z)
$4.80e4s - 4.83e4s$	$5.54m/6.05m/6.57m$
$4.83e4s4.87e4s$	$101.86m/138.96m/27.36m$
$4.87e4s4.91e4s$	$157.20m/123.56m/58.78m$

Table 2.3: The average error of the velocity in different time segments

2.4 the accelerometer and gyroscope measurements are shown. As expected do we have highest acceleration in x direction and the fastest rotational speed in z direction. In the accelerometer x axis and the gyroscope z axis measurement can be seen the 5 steps of different rotational speed. The very first step is a test rotation followed by the hit on the sensors which react with a peak. Since the sensitivity of

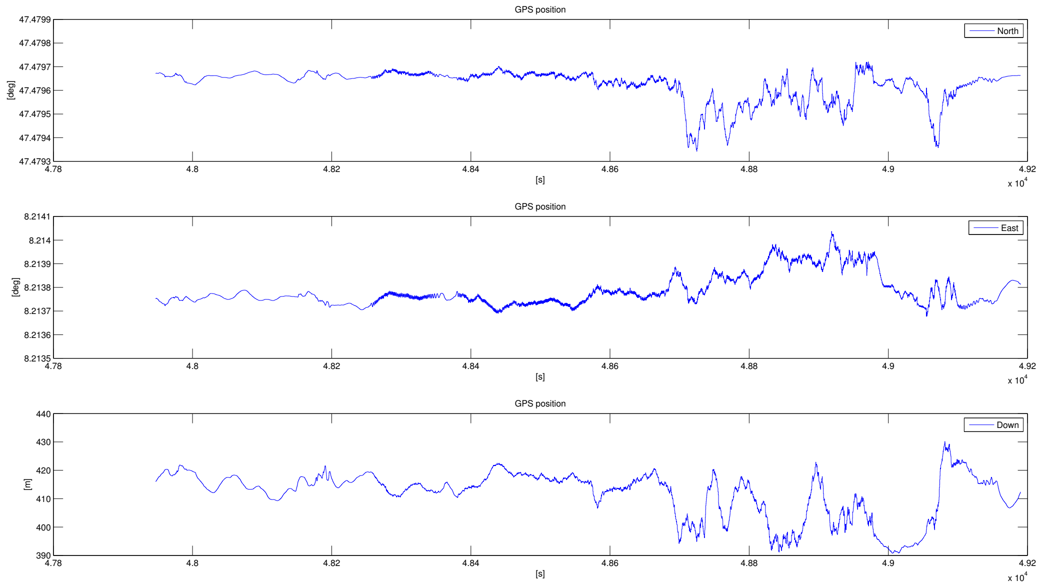


Figure 2.1: GPS Position

the PX4 accelerometer is set to $+/-4g$, the output of accelerometer in x direction is limited to $39.24m/s^2$. The acceleration in y direction is shows small steps due to the placement of the sensors slightly besides the center of the box.The gyroscope from the XTi-G shows a limitation in measuring more than $403deg/s$. The x-axis and y-axis should be zero for the gyroscope since the rotation is in two dimensional. Obviously is the noise again increasing with a higher rotation speed.This can be clearly seen in the acceleration in z direction which should be constant at $-9.81m/s$, the gravitation. It looks like the noise level of the PX4 is the highest before the x-IMU and the MTI-G which has the lowest. This can be traced back on the high sensitivity set up of the PX4 accelerometer.

In figure **??** the magnetometer output is plotted. There is a miss alignment between the x-IMU,MTi-G and the PX4 which cannot be explained by the authors. The range of the maximum value and the minimum value of the PX4,MTi-G and the x-IMU is in x and y direction $0.4gauss$. This is the expected range in the region of Zurich. In this sense, if the the output is added with an offset in order to get the range to $+/-0.2gauss$, the measurements should be correct. The PX4 has a mean value of 0.3893 gauss while the ground truth is $0.4268gauss$. The noise of the magnetometer is in the x-IMU higher than in the other two IMUs.

In figure 2.6 a closer look on the x axis of the magnetometer and accelerometer and on the y axis of the gyroscope is presented. The form of the curve has sine behavior, what we would expect again.

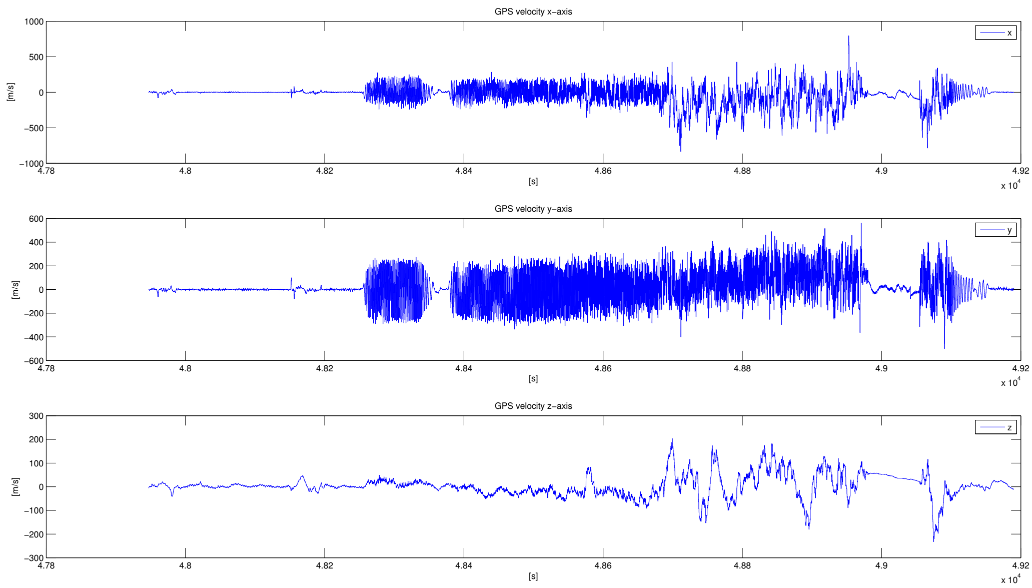


Figure 2.2: GPS velocity

2.1.3 Conclusion

All IMUs work in the same range of noise level. With an increasing rotational velocity and therefore a higher centripetal acceleration, the average error is increasing in all sensors, except in the magnetometer. There could not be found a noise acceleration dependency. In a next experiment also the recording of rotational velocity should be carried out in order to generate a ground truth to compare the it with the IMU datas. It can then be made some more observations to compare the quality of the sensors with each other. Finally the sensitivity of the accelerometer of the PX4 unit should be set in a way to be able to observe the whole range or the applied and tested centripetal acceleration.

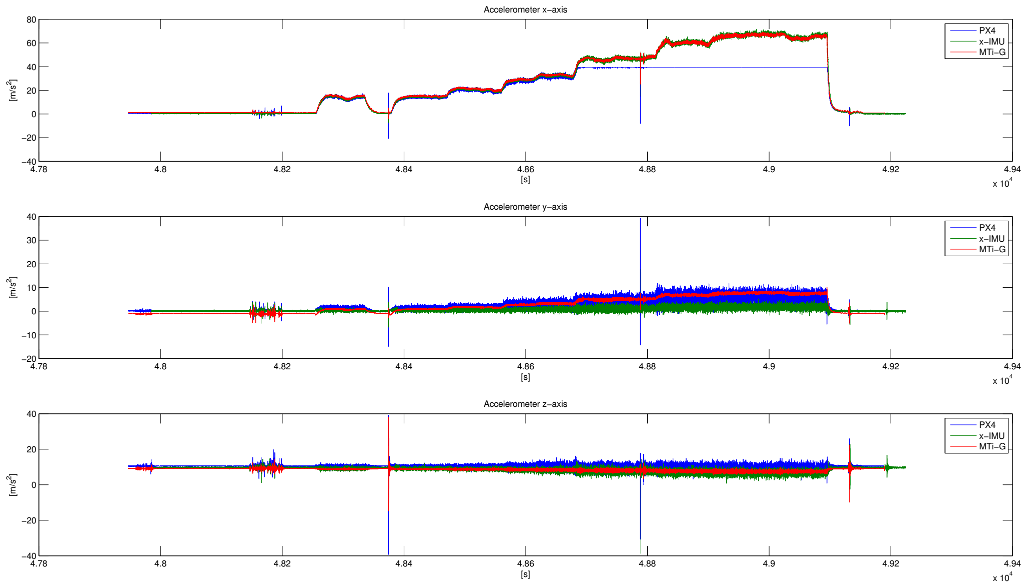


Figure 2.3: Accelerometer

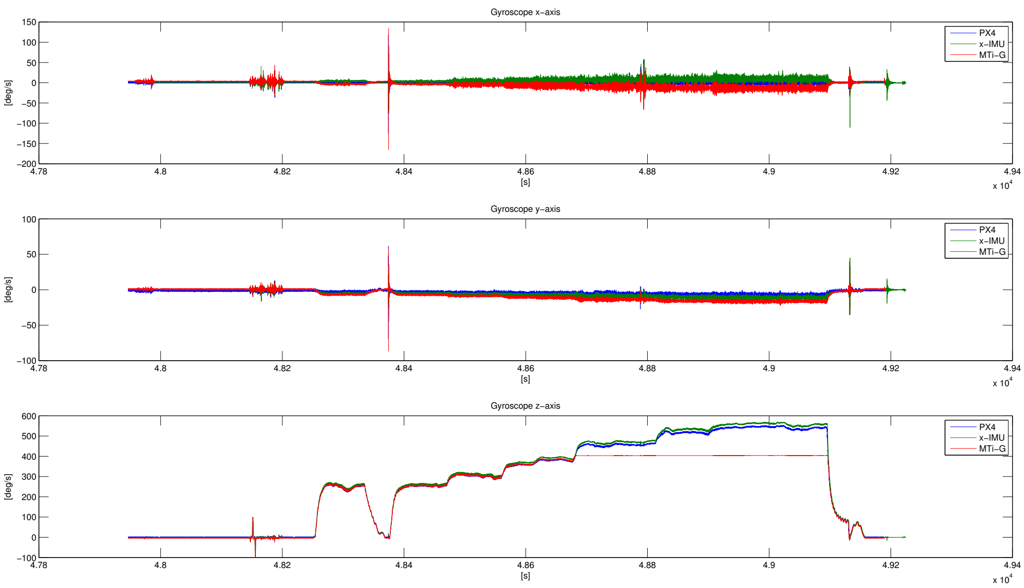


Figure 2.4: Gyrometer

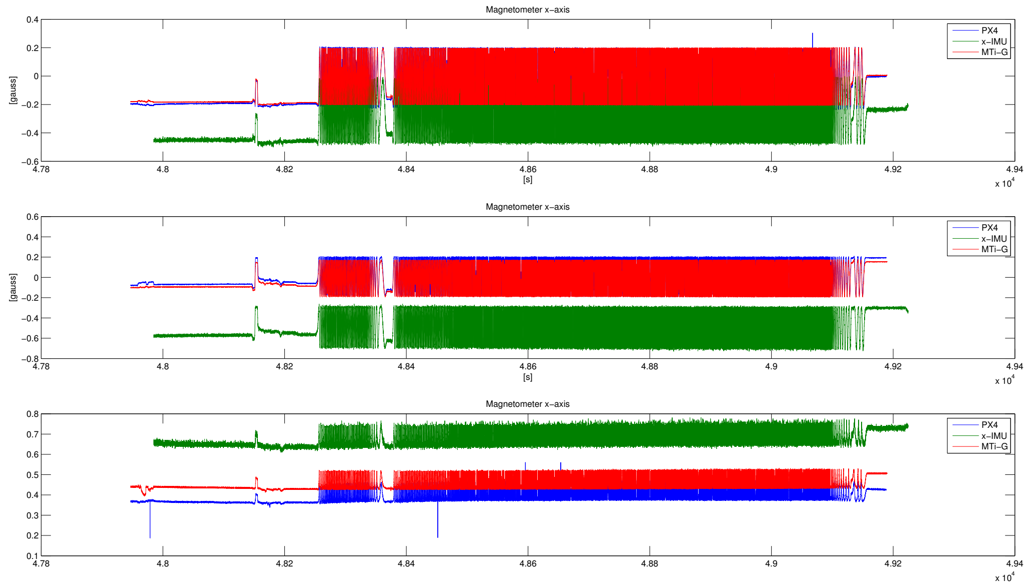


Figure 2.5: Magnetometer

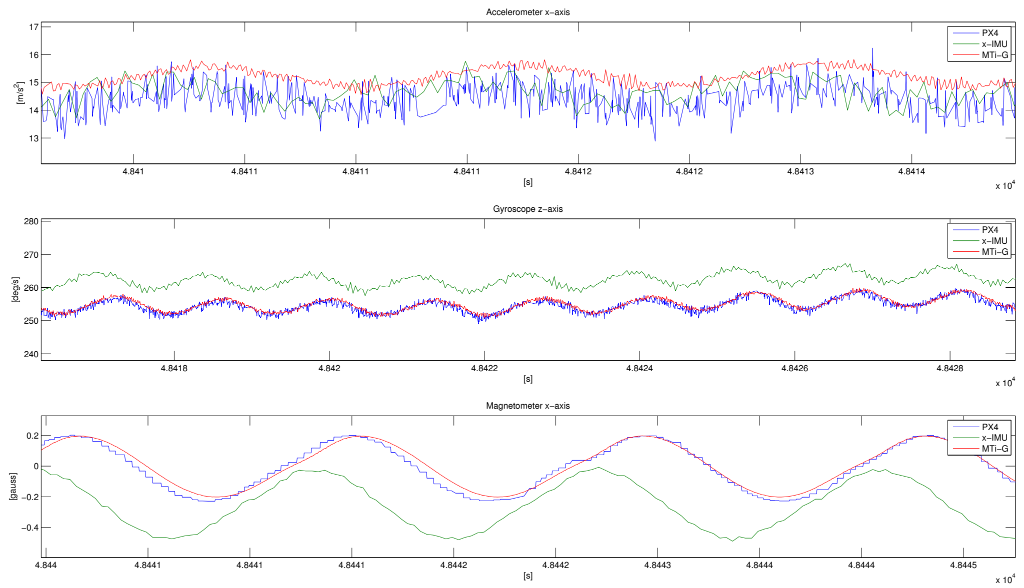


Figure 2.6: A closer look in time on the three IMU sensors.

Chapter 3

State Estimation

In this chapter is first an introduction in the Kalman Filter and the modified version called Extendend Kalman Filter given. In a second step will the state estimation algorithm, developed during this thesis explained. This gives an overview over the whole code and describes how the parameters were set and how a sensor data outage is handled. Section 3.3 and 3.4 will focus on the two essential parts of the algorithm, how the states are estimated and how the measurements are estimated.

3.1 Extended Kalman Filter

Discrete Kalman Filter

We will first give a short introduction to the Kalman Filter in general and its mathematical justification.

> The Kalman filter is an extremely effective versatile procedure for combining noisy sensor outputs to estimate the state of a system with uncertain dynamics.

[âĂęangu...] It was published in 1960 by R.E.Kalman. The Filter gained a strong impact in the area of autonomous or assisted navigation [..pap...]. Systems in which the filter is estimating the state $x \in R^n$ can be described by the following linear stochastic difference equation.

$$x_{k+1} = A_k * x_k + B_k * u_k + w \tag{3.1}$$

The next equation shows the relation between the state and the measurement $z \in R^n$.

$$z_k = H_k * x_k + D_k * u_k + v \tag{3.2}$$

Where the random variables w and v stand for the process and the measurement noise, which are assumed to be independent, white and normal probability distributions.

> The $n \times n$ matrix A in equation 3.1 relates the state at time step k to the state at step $k+1$, in the absene of either a dirving function or process noise. The $n \times l$ matrix B relates the control input $u \in R^l$ to the state x. The $m \times m$ matrix H in the measurement equation 3.2 relates to the state to the measurement z_k.

[.pap.] In most navigation applications, the noisy sensors are a GPS receiver and an inertial navigation system or other sensors like for example speed sensors. The states of a system are described by the position, velocity, attitude and attitude rate [.angu..].

Mathematical Background

In figure 3.1 the structure of the algorithm is sketched for a better understanding of the filter's mathematical background. In a first step the state at $k+1$ is estimated

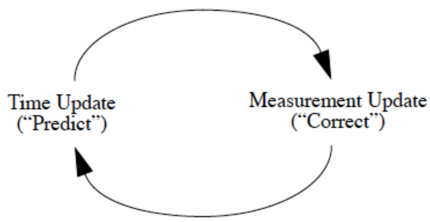


Figure 3.1: A simple structure of the algorithm

based on 3.1. This estimation is called a priori state estimation and written as $\hat{x}_k^-$. In a second step the a priori state estimation is corrected by the knowledge of measurements z_k. This corrected estimation is called the a posteriori estimation and written as $\hat{x}_k$.

Now two error can be defined with it's a error covariances. First the error of the a priori estimation

$$e^- = x_k - \hat{x}_k^- \tag{3.3}$$

and second the error of the a posteriori estimation

$$e = x_k - \hat{x}_k \tag{3.4}$$

with the a priori estimate error covariance

$$P_k^- = E[e^- e^{-T}] \tag{3.5}$$

and the a posteriori error covariance.

$$P_k = E[ee^T] \tag{3.6}$$

The Kalman Filter is calculating the a posteriori estimation, the one we are finally looking for. This calculation is a linear combination between the a priori estimated state and the difference of the the estimated measurement $x_k * H$ and the actual measurement z_k weighted with the Kalman gain K. This difference $z_k - H_k \hat{x}_k^-$ is called residual and tells us how accurate the estimation of the measurements is. This is summarized in equation 3.7.

$$\hat{x}_k^- * K * (z_k - H_k \hat{x}_k^-) \tag{3.7}$$

The Kalman gain K is the heart of the Kalman Filter. K, a $n \times n$ Matrix is chosen in a way to minimize the a posteriori error covariance shown in equation **??**. With some calculations the following

$$K = \frac{P_k^- * H_k^T}{(H_k * P_k^- * H_k^T + R_k)} \tag{3.8}$$

can be derived for the Kalman gain K.[..pap..]] This concept uses the information of a normal gaussian distribution of the noise mentioned in (..equ2..). P_k is minimized using the Maximum Likelihood Estimation and is called the Gaussian Maximum-Likelihood Estimator [âĂę]. A more detailed derivation of the mathematics is shown in [..maybeck..] and [..angu..].

Algorithm

With equations described in the section above, the estimation and correction step can be summarized as shown in figure 3.2. For a fine tuning of the filter often the

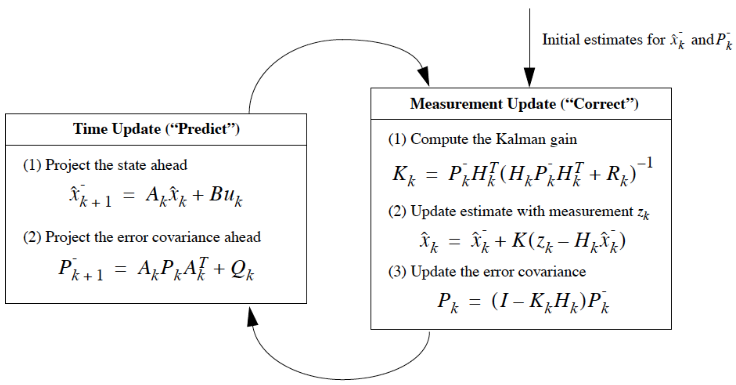


Figure 3.2: The prediction and the correction step with their corresponding equations.

measurement error covariance matrix R_k and process noise Q_k can be used. R_k contains the information how accurate the sensors and therefore the measurements are expected to be. This can be estimated in a first round with some static sensor test or can be found in the data sheets of the sensors. The matrix Q_k describes how correct the propagation model described by the matrix A_k is believed to be. In this sense as higher the values and therefore the noise of R_k and Q_k the less influence do the measurements or the propagation model respectively have on the estimated state. Having a diagonal matrix with constant values over time, does not represent the reality, but brings it to a form which is fast stable and easier tunable[..pap..].

Extended Kalman Filter (EKF)

In reality the process to estimate the new state and the measurement relationship to the process is non-linear.

$$x_{k+1} = f(x_k, u_k, w) z_k = h(x_k, v_k) \tag{3.9}$$

One of the solution to overcome this problem is the Extended Kalman Filter shortly EKF. The EKF is linearizing the this functions with a first order Taylor-Series around the current state. This can be summerized with the following equation:

$$x_{k+1} \approx \hat{x}_{k+1} + A(x_k - \hat{x}_k) + w \qquad z_k \approx \hat{z}_k + H(x_k - \hat{x}_k) + v \tag{3.10}$$

Where A and H are the Jacobian matrix of the function f and h.

$$A_{i,j} = \frac{\partial f_i}{\partial x_j}(\hat{x}_k, u_k, 9) \qquad H_{i,j} = \frac{\partial h_i}{\partial x_j}(\hat{x}_k, u_k, 9) \tag{3.11}$$

This brings us to the summarized equations of the EKF in figure 3.3.

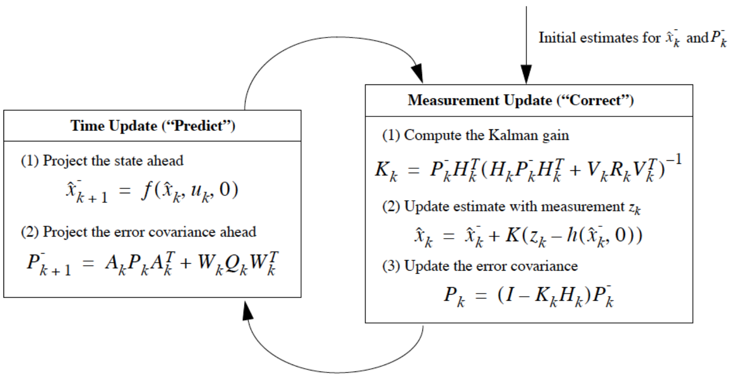


Figure 3.3: The prediction and correction step of the EKF with their corresponding equations.

3.2 Algorithm for State Estimation

In this chapter we will present the structure of the state estimation algorithm as we implemented it in MATLAB. As mentioned before, the goal is to get an estimation of the system's state as precise as possible using an Extended Kalman Filter. The Algorithm can be split in several main tasks. First the physical relations between the state x_k and the state x_{k+1} has to be defined and summarized in a propagation model. This is needed to calculate the a priori state estimation as described in chapter 3.1. Since this step is crucial for the accuracy of the algorithm, it is further explained in chapter 3.3. A second key role plays the measurement estimation which corrects the a priori state estimation and calculates the a posteriori state estimation. A more detailed explanation of this correction can be found in chapter 3.4. A third question to be elaborated is how the algorithm has to behave if no sensor data is available or if not all sensors provide their outputs with the same frequency. In the last paragraph we will explain how the error covariances were chosen.

Orientation

The filter has to handle two different coordinates frames. One is the inertial frame and the other the body frame. The inertial frame is the global frame. The position and the velocity both provided by the GPS are in the inertial frame. The origin of the inertial frame lies at the suspension point of the pendulum/kite. The body frame is a coordinate system on the kite. Its z-axis always points in the direction of the line and its origin lies at the center of gravity of the kite. The sensor measurements from the IMU are all in the body frame. Figure 3.4 contains an illustration of these two coordinate systems.

The euler angles describe how the inertial frame has to be rotated to bring into the body frame. With other words, the euler angles tell us the orientation of the body frame. With the direct cosine matrix (DCM) we can transform a vector from the inertial frame into the body frame and vise versa. For example can the accelerometer data which is given in body frame coordinates be transformed into the inertial frame, where the propagation of the states is calculated.[..angus..]

Degrees of Freedom

As stated in the introduction, we implemented two different versions of the state estimation algorithm. A standard implementation assuming a free mass and one version that exploits the constraints given by the knowledge that the mass is oscillatingly suspended at a fixed point. These two versions are quite similar and share a lot of code, but one crucial point where they differ from each other is the number of degrees of freedom (DoF). The free mass model uses 12 DoF, namely position, velocity, attitude and attitude rate. Thus the state vector consists of 12 variables. The pendulum model uses only 6 true degrees of freedom: Three angles representing the position and attitude as well as their respective rates representing velocity and

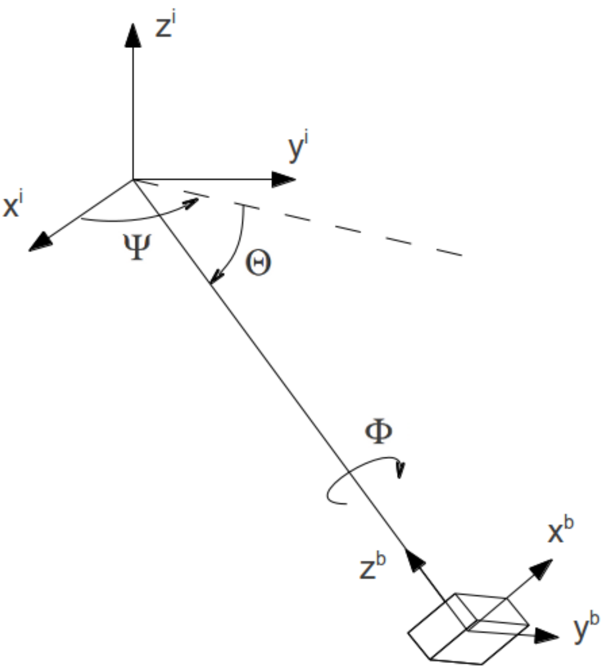


Figure 3.4: The definition of the inertial and body frame as well as the convention for the angles in the pendulum model.

attitude rate. The definition of these angles can be seen in figure 3.4.

States and Measurements

The state vector in the free mass model with the dimension of 12 looks as follows:

$$x = \begin{pmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \phi \\ \theta \\ \psi \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \tag{3.12}$$

The first six variables are position and velocity in the inertial frame. The angles and angular rates are Euler angles with the convention ZY'X". This means that in order to get from the inertial frame to the body frame the following three rotations have to be performed: a rotation with angle ψ around the z-axis in the inertial frame, a rotation with angle θ around the new y-axis and a last rotation with angle ϕ around the new x-axis.

The state vector in the pendulum model has eight elements. Even tough only 6 DoF are assumed, two additional variables, the radius and the change in radius, are included in the state to ensure the possibility of loosening the constraint of a fixed line length and implementing a spring model as suggested later on in the outlook section. Thus the state vector is defined as follows:

$$x = \begin{pmatrix} \Phi \\ \Theta \\ \Psi \\ r \\ \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \\ \dot{r} \end{pmatrix} \tag{3.13}$$

For the definition of the angles Φ, Θ and Ψ see figure 3.4.

The measurement vector is the same for the free mass model as well as for the pendulum model. It consits of the position (pos) and the velocity (vel) of the GPS, the acceleration (acc) from the accelerometer and the rate of turn (gyro) from the gyroscope and the orientation (mag) from the magnetometer. Each of them in all three dimensions give us the measurement vector z with a dimension of 15.

$$z = \begin{bmatrix} pos\ x \\ pos\ y \\ pos\ z \\ vel\ x \\ vel\ y \\ vel\ z \\ acc\ x \\ acc\ y \\ acc\ z \\ gyro\ x \\ gyro\ y \\ gyro\ z \\ mag\ x \\ mag\ y \\ mag\ z \end{bmatrix} \tag{3.14}$$

Structure of the Algorithm

In 3.5 the schematic of the algorithm is shown. Each of the blocks will now successive explained.

Block 1, totalTime: The filter estimates with constant rate the state of the system. This rate is independent on any output rate of the sensors. In the first block the time at which the state is estimated is set. It is the old time totalTime added with the constant time step t.

Block 2, Propagation: During the second block the a priori state of the system is estimated with the help of the physical model. How and why the physical model is defined is explained in chapter 3.3. After this step the vector x_{est} is up to date.

Block 3, measurement availability: If there is no new measurement until the time of the a priori estimated state, the correction step is not executed. This allows the algorithm to run the ekf with a higher rate, then the IMU provide data. To summarize: If we have no new data from the sensors the system just keeps propagating based on the physical model.

Block 4, finding the closest measurement: In this block it is searched for the closest measurement to the time of the a priori estimated state in block 2. If until the time of the estimation several sensor values are available only the newest and therefore most accurate value is taken for the correction.

Block 5, measurement selection: Mostly not all sensors of an IMU provide an output with the same rate. Here it is tested, weather we have a new sensor

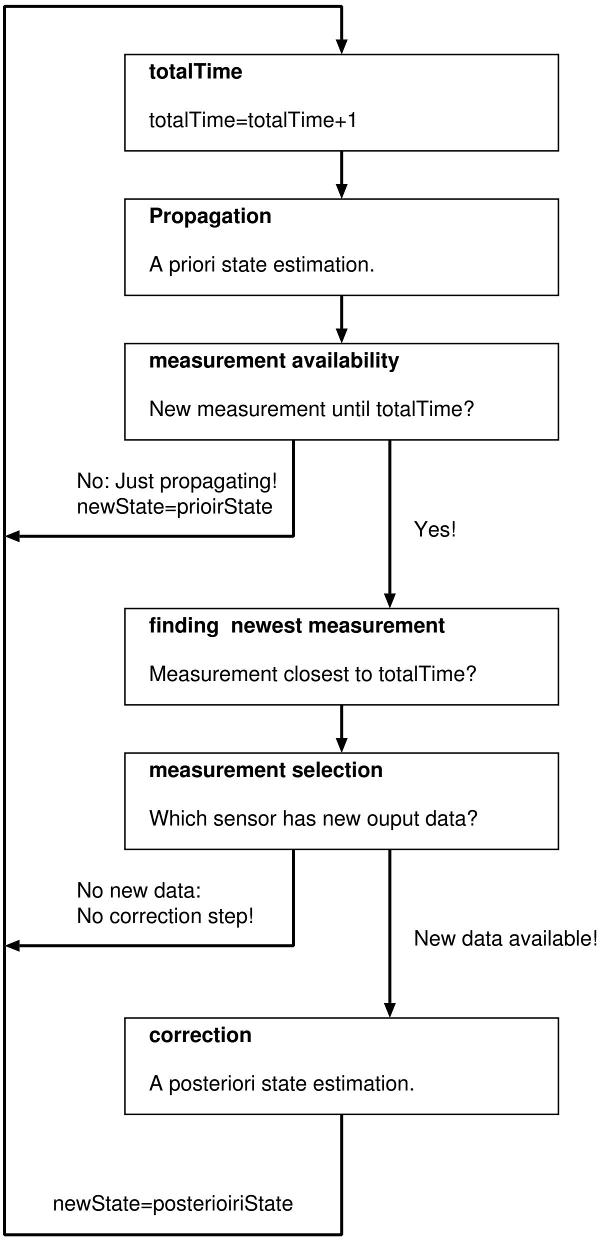


Figure 3.5: The graphical represenetation of the algorithm's structure.

value or is it still the one of the previous estimation. The correction step in block 5 is only for the new data executed. The other one still just propagate.

Block 6, correction: As last step the correction of the a priori estimation takes place and the posteriori estimation is calculated. How the relationship between the state and the measurements look like is explained in chapter 3.4

Error Covariance

For the measurement error covariance the noise of the different sensors are taken from the data sheets. But they were manually scaled according to have more accurate and stable filter. For the propagation model the covariance was estimated on how accurate the equations are. But also in this case were they manually adjusted afterwards. For an easier handling only the diagonal elements have a value. The off diagonal values are all zero.

Jacobian Matrix

In chapter 3.1 the matrixes H and A were derived. Since the physical model and relations between state and the measurements are non linear, H and A are the Jacobian matrixes of the functions f and h. For having an accurate as possible solution the Jacobian was in a first version calculated analytically with symbolic toolbox from MathWorks. The Jacobian matrix then has to be evaluated in every iteration step. Simulating 0.01 seconds took the filter much longer than 10 minutes. It was then decided to rewrite the algorithm calculating the Jacobian matrixes numerical. The function ekf.m [..ekfwebsite..]from MathWorks was restructured to match the requirements of this algorithm. It uses a complex step differentiation to calculate the derivatives of the function f and h [..papCompDiff..].

3.3 State Estimation Model

In order to propagate the state of the system x_k to the a priori state estimation x_{k+1}^- the filter needs to be able to predict in what state the system will be at time t_{k+1} based on its state at time t_k. In the most basic case, the assumption is that there exists no knowledge about the forces acting on the body. This leads to the so called ”free mass model”. However, in the case of a kite tethered to a ground station, the body is not able to move freely in three dimensions, but its movement is in a first approximation limited to a sphere with a given radius. This means, that one of the main forces acting on the body, the centripedal force, can be accounted for in the prediction. Furthermore an aerodynamical model of the kite might be able to predict also other forces than the centripedal force. In order to explore the benefits of such a model, we decided to use a spherical pendulum for the resons described in the introduction. In that case, all the forces acting on the body are known and the degrees of freedom are further reduced due to the bridled suspension. In the following sections, these two models are going to be described in detail.

Free Mass Model

The dynamics of the body are described by a system of first order differential equations describing the change in the state variables.

$$\dot{x} = g(x, t) \tag{3.15}$$

In this case the model is time invariant and thus g is not time dependant.

$$\dot{x} = g(x) \tag{3.16}$$

Since there is no knwoledge about the forces acting on the body, this set of equation looks rather simple:

$$g(x) = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ 0 \\ 0 \\ 0 \\ x_{10} \\ x_{11} \\ x_{12} \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.17}$$

Pendulum

Again the dynamics are described by a system of time invariant first order differential equations.

$$\dot{x} = g(x) \tag{3.18}$$

Due to the gravitation and the centripedal force being the only forces acting on the body, we can derive an exact set of differential equations that govern the motion of a spherical pendulum. Therefore we use the Euler-Lagrange-Equation to derive the differential equations as suggested in [12, 156ff].

With the kinetic energy

$$T = \frac{1}{2} m r^2 (\dot{\Theta}^2 + \dot{\Psi}^2 \cos^2 \Theta) \tag{3.19}$$

and the potential energy

$$V = -mr \sin \Theta \tag{3.20}$$

the Lagrangian is defined by:

$$L = T - V = \frac{1}{2}mr^2(\dot{\Theta}^2 + \dot{\Psi}^2\cos^2\Theta) + mr\sin\Theta \tag{3.21}$$

Substituting the Lagrangian into Euler-Lagrange-Equation

$$\frac{\partial L}{\partial \Psi} - \frac{d}{dt}\frac{\partial L}{\partial \dot{\Psi}} = 0 \tag{3.22}$$

and solving for $\ddot{\Psi}$ results in:

$$\ddot{\Psi} = 2\tan\Theta\dot{\Theta}\dot{\Psi} \tag{3.23}$$

And similarly for $\ddot{\Theta}$:

$$\frac{\partial L}{\partial \Theta} - \frac{d}{dt}\frac{\partial L}{\partial \dot{\Theta}} = 0 \tag{3.24}$$

$$\ddot{\Theta} = \frac{\cos\Theta(g - r\sin\Theta\dot{\Psi}^2)}{r} \tag{3.25}$$

The third degree of freedom, the body's rotation about its z-axis, is assumed to be constant. However, due to the convention of the euler angles, a change in phi contributes an additional term to the z component of the rotation vector in the body frame. (For further explanation see section ...) To compensate for that, the second derivative of Φ is not zero but defined as follows:

$$\frac{d}{dt}w_z^{body} = \frac{d}{dt}(\dot{\Phi} - \sin\Theta\dot{\Psi}) = 0 \tag{3.26}$$

$$\ddot{\Phi} = \cos\Theta\dot{\Theta}\dot{\Psi} + \sin\Theta\ddot{\Psi} \tag{3.27}$$

With the radius assumed to be constant, the fourth and eighth state variable (r and $\dot{r}$) are not changing and thus their time derivatives are zero.

Therefore the set of differential equations is given by:

$$g(x) = \begin{bmatrix} x_5 \\ x_6 \\ x_7 \\ 0 \\ \cos x_2\dot{x_2}\dot{x_3} + \sin x_2\ddot{x_3} \\ \frac{\cos x_2(g - r\sin x_2\dot{x_3}^2)}{r} \\ 2\tan x_2\dot{x_2}\dot{x_3} \\ 0 \end{bmatrix} \tag{3.28}$$

Solving the Differential Equations

So far we have derived the sets of differential equations that contain all the knowledge we have about the systems. To calculate the a priori state estimation x^-_{k+1} from the state of the system x_k we now only need to solve these sets of differnetial equations. This is done numerically using the classical fourth-order Runge-Kutta method [13]. Note that in our case g(x) is time invariant.

$$k_1 = hg(t_n, x_n) \tag{3.29}$$
$$k_2 = hg(t_n + \frac{1}{2}h, x_n + \frac{1}{2}k_1) \tag{3.30}$$
$$k_3 = hg(t_n + \frac{1}{2}h, x_n + \frac{1}{2}k_2) \tag{3.31}$$
$$k_4 = hg(t_n + h, x_n + k_3) \tag{3.32}$$
$$x_{n+1} = x_n + \frac{1}{6}k_1 + \frac{1}{3}k_2 + \frac{1}{3}k_3 + \frac{1}{6}k_4 + \mathcal{O}(h^5) \tag{3.33}$$

3.4 Sensor Estimation Model

An illustration of the relation between the position and velocity in (y, x, z) or (dx, dy, dz) and the two angles Φ, Θ and Ψ is shown in figure 3.4. With this sketch, the assumption of a constant radius and the fact that the velocity is the derivative of the position we get to the equations:

$$pos\ x = R * cos(\Theta)cos(\Psi) \tag{3.34}$$
$$pos\ y = R * sin(\Psi)cos(\Theta) \tag{3.35}$$
$$pos\ z = R * sin(\Theta) \tag{3.36}$$
$$vel\ x = -sin(\Theta)cos(\Psi) * R * \dot{\Theta} - cos(\Theta)sin(\Psi) * R * \dot{\Psi} \tag{3.37}$$
$$vel\ y = -sin(\Theta)sin(\Psi) * R * \dot{\Theta} + cos(\Theta)cos(\Psi) * R * \dot{\Psi} \tag{3.38}$$
$$vel\ z = -cos(\Theta) * R * \dot{\Theta} \tag{3.39}$$
$$\tag{3.40}$$

The centripetal acceleration always accting in z direction of the sensor can be calculated by the formula:

$$a_{cp} = \frac{v^2}{R} \tag{3.41}$$

We then have the an additional acceleration of the earth gravitation. The earth gravitation is in the inertial frame only in z direction and has to be transformed with the DCM_{bi} from the inertial in the body frame. Finally the acceleration can be written as:

$$acc = \begin{bmatrix} 0 \\ 0 \\ \frac{v^2}{R} \end{bmatrix} + DCM_{bi} * \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.42}$$

The magnetic field is a constant value in Zurich. By transforming that into the body frame again with the help of the DCM_{bi}, the expected measurement from the magnetometer is calculated:

$$mag = DCM_{bi} * \begin{bmatrix} 0.2145 \\ 0.0060 \\ 0.4268 \end{bmatrix} [gauss] \tag{3.43}$$

The gyroscope measurement is directly represented by the rate of the euler angles Φ,$\dot{\Theta}$ and $\dot{\Psi}$. Since the euler angular rate has to be transformed into the body frame coordinates and additinally has a converstion about which axes is rotated the first, what the gyroscope has not,a transformation matrix has to be used as follows:

$$gyro = \begin{bmatrix} 1 & 0 & -sin(\Theta) \\ 0 & cos(\Phi) & sin(\Phi)cos(\Theta) \\ 0 & -sin(\Phi) & cos(\Phi)cos(\Theta) \end{bmatrix} * \begin{bmatrix} \dot{\Psi} \\ \dot{\Theta} \\ \dot{\Phi} \end{bmatrix} \tag{3.44}$$

Because the position of the senors is not at the center of mass of the pendulum, an additional displacement vector has to be added to the position (compare figure 3.6). With transforming the distance between the center of mass and the sensor into the inertial frame, the position is adjusted. The displacement gives an additional velocity which is the crossproduct of the rate of rotation and the displacement from the center of mass. By the derivative of this additional velocity the additional acceleration is calculated. The displcement has no influence on the gyrometer measurement and the magnetometer.

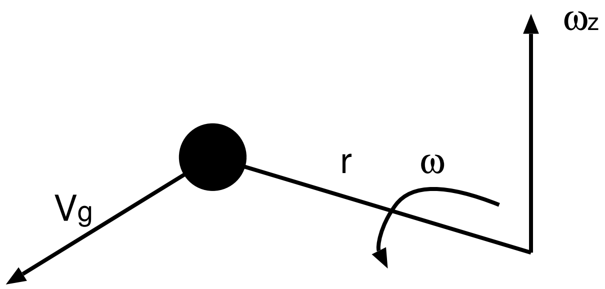


Figure 3.6: The black dot represents the sensor, while r is the distance between the center of mass of the pendulum and the sensor.

Chapter 4

Results - Vicon Test

In this chapter the algorithm from chapter 2 is tested under different conditions. In a first part (see chapter 4.1) the set up of the test is presented. In the second part (see chapter 4.2). The different test conditions are described and the results shown.

4.1 Set up

To verify the performance of the kalman filter algorithm described in chapter 2 a ground truth is needed. By using the motion capture system from the english company vicon some test were performed. The vicon system provides the position of the sensors and the orientation. Because this test has to be performed indoors no GPS signal was available. The vicon data for position added with some noise was simulated as GPS measurement of ekf and the derivative of the vicon's position added with some noise was pretended to be the velocity provided by the GPS. The impact of the noise can be seen in chapter 4.2.1. The pendulum was set as it can be seen on the picture in figure 4.1. The box with the two sensors PX4 and MTi-G is showed in figure (...). How the ekf state estimation differs depending weather the measurements are used from the PX4 IMU or from the Mti-G IMU can be read in chapter 4.2.3. In addition to that a kalman filter which does not take into account the information of a pendulum's movement has to be implemented. A free mass model state estimation is used as control algorithm. The varying results between the state estimation with using the pendulum's physical behavior and without are shown in chapter (..).

4.2 Evaluation

Four comparisons are made to get an overview of the estimators performance. This chapter shows how different test conditions influence the performance. In chapter 4.2.1 the estimator is tested once with a low noise GPS and one with a higher noise GPS signal. Since a GPS is not always received or has at least a lower frequency than the IMU measurements, the behavior of the estimator without GPS is a main

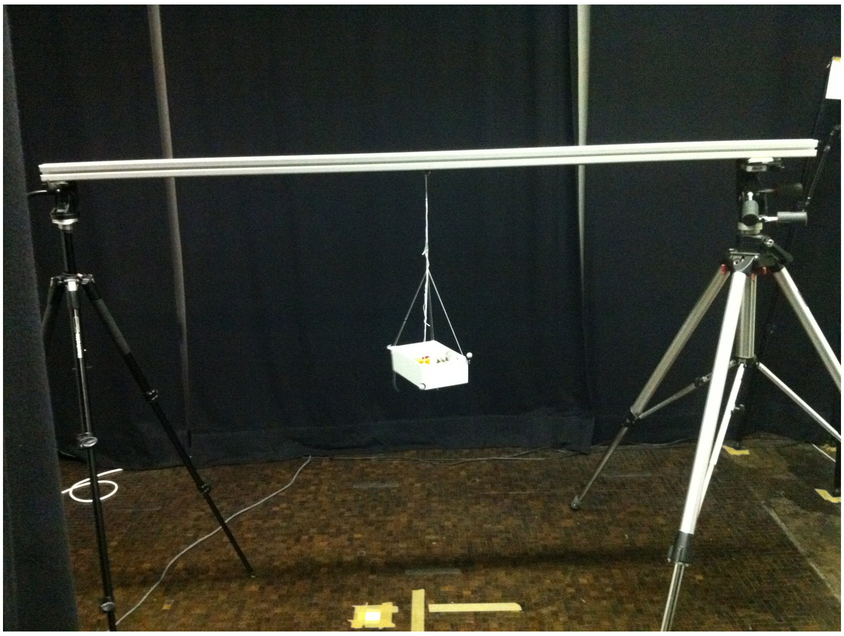


Figure 4.1: Pendulum set up.

performance criteria for an estimator. In chapter 4.2.2 this condition is tested. How the measurements of different IMUs are influencing the estimator is shown in chapter 4.2.3. Of course it has to be verified if an estimator using a the information of the pendulums physical behavior is more accurate than a estimator not using this information. The results of this question can be seen in chapter **??**.

4.2.1 Noise of the GPS

The quality of the received GPS signal can vary according to different circumstances. The weather conditions or an acceleration as we have seen in chapter 2 can have an influence on the signals quality. As mentioned before is the GPS signal in the vicon test only simulated according the vicon data. This has the advantage of being able to use a GPS with different SNR since this is artificially added on the vicon's position data. In figure **??** an SNR of 5 dB is added and in figure 4.4 a SNR of 25 dB is added. The estimator somehow differs but, the noise level does not have a strong impact. In the estimator are the noise covariance matrixes adjust depending on which SNR the signal has. In the first case of a 5 dB SNR the noise of the GPS position and velocity are defined as 0.001 and 0.01 respectively. In the second case when having a noise level of 25 dB the noise in the estimator is increased to 0.01 for the position and 0.05 for the velocity. In figure 4.5 and figure 4.6 the errors in position and orientation are plot for the 5 dB and 25 dB GPS signal respectively. The error is calculated by the norm of the difference between estimated position

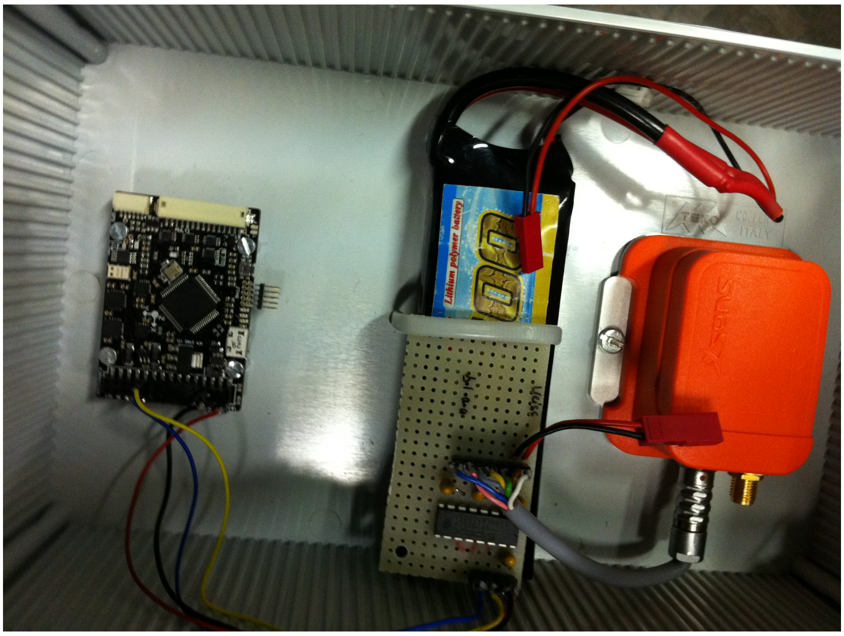


Figure 4.2: Box with sensors.

and the effective position:

$$error = \left| \begin{bmatrix} vicon_{pos\ x} \\ vicon_{pos\ y} \\ vicon_{pos\ z} \end{bmatrix} - \begin{bmatrix} state_{pos\ x} \\ state_{pos\ y} \\ state_{pos\ z} \end{bmatrix} \right| \tag{4.1}$$

For the orientation only the error of the first angle is calculated since the other two depend on the first one. When calculating the mean value of the error, starting at the point, when the GPS signal is settled down, following results are obtained:

$$meanError_{pos}^{5\ dB} = 0.0162\ m \tag{4.2}$$

$$meanError_{pos}^{25\ dB} = 0.0223\ m \tag{4.3}$$

$$meanError_{\Theta}^{5\ dB} = 0.1440\ rad \tag{4.4}$$

$$meanError_{\Theta}^{25\ dB} = 0.1584\ rad \tag{4.5}$$

And a ration of:

$$\frac{meanError_{pos}^{5\ dB}}{meanError_{pos}^{25\ dB}} = 0.7265 \tag{4.6}$$

$$\frac{meanError_{\Theta}^{5\ dB}}{meanError_{\Theta}^{25\ dB}} \quad 0 \quad 0.9091 \tag{4.7}$$

4.2.2 GPS outage

Since explained before is the performance of a estimator without having a GPS signal an important criteria. In figure **??** a detail of the plot is shown. The estimator

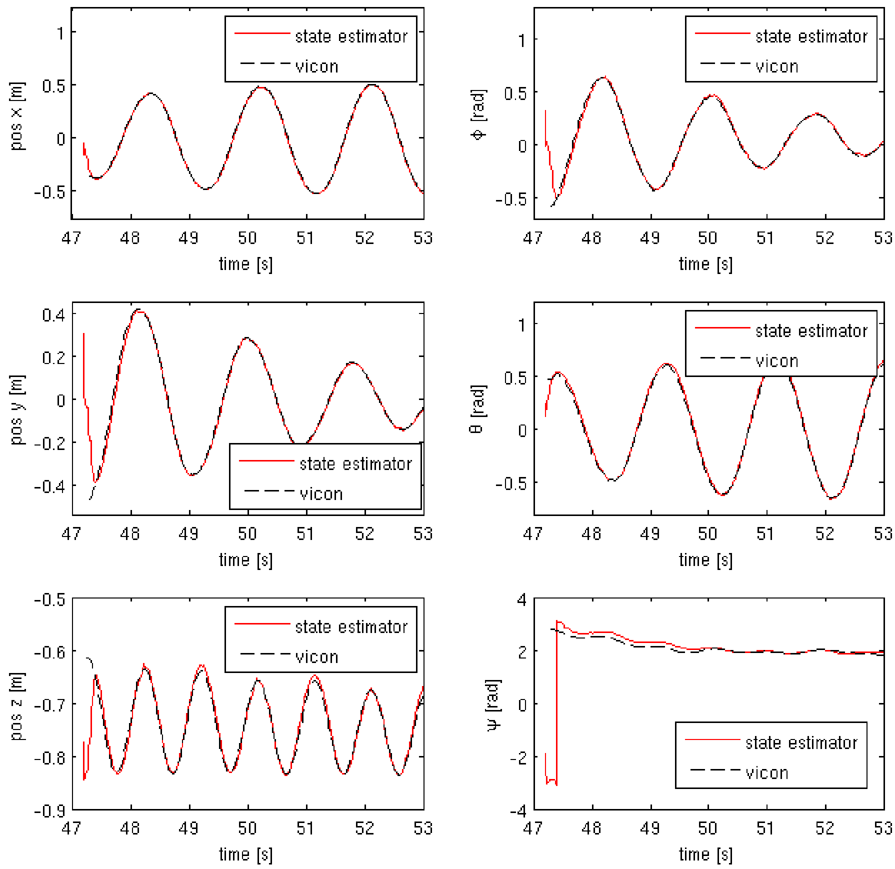


Figure 4.3: GPS signal with a SNR of 5 dB.

is able to estimate the state only by using the IMU measurements. In this case the MTi-G data is used. Again can the error of the estimated position and the true position be calculated and plot as well as the difference between the estimated angle of Θ and the true angle as it is done in figure (..). The mean value of this error is calculated in the same way as the error between the different noise levels in chapter 4.2.1: The ratios of error of the estimation with and without GPS are shown in equation:

4.2.3 The MTi-G unit versus the PX4 unit

In this section the influence of different measurements on the estimator are presented. The above test of the estimator without having a GPS signal is carried out again with the PX4 measurements. In figure **??** the same detail is shown. The

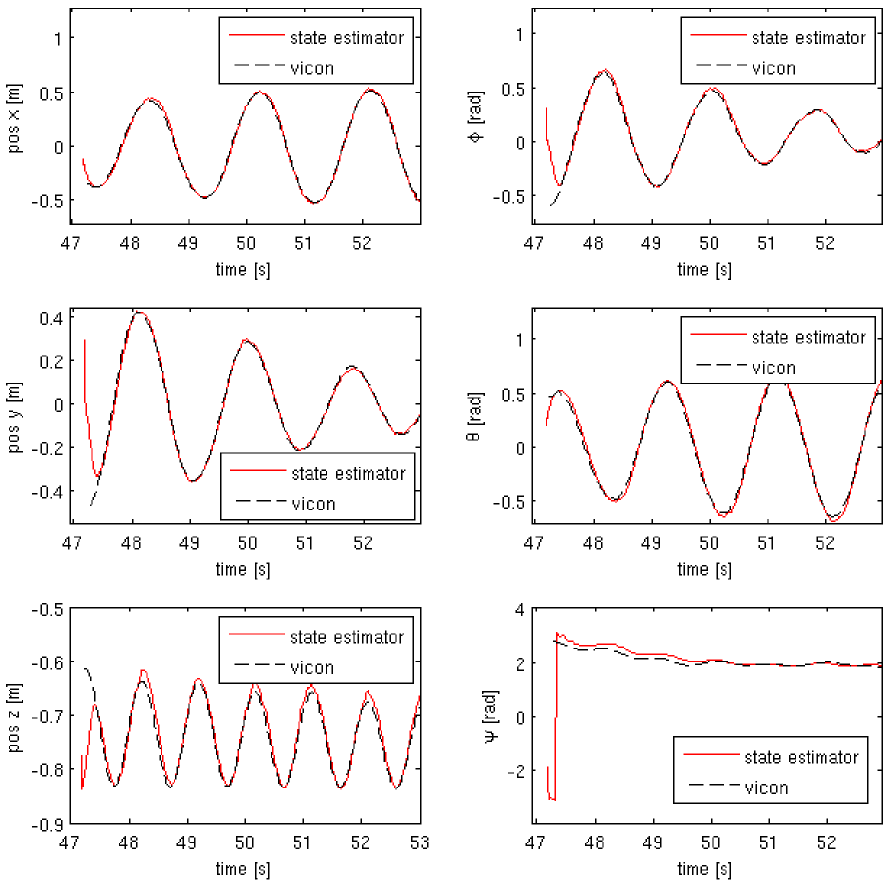


Figure 4.4: GPS signal with a SNR of 25 dB.

error is plot in figure 4.9. The calculation of the ratios shows the big impact of the measurement unit.

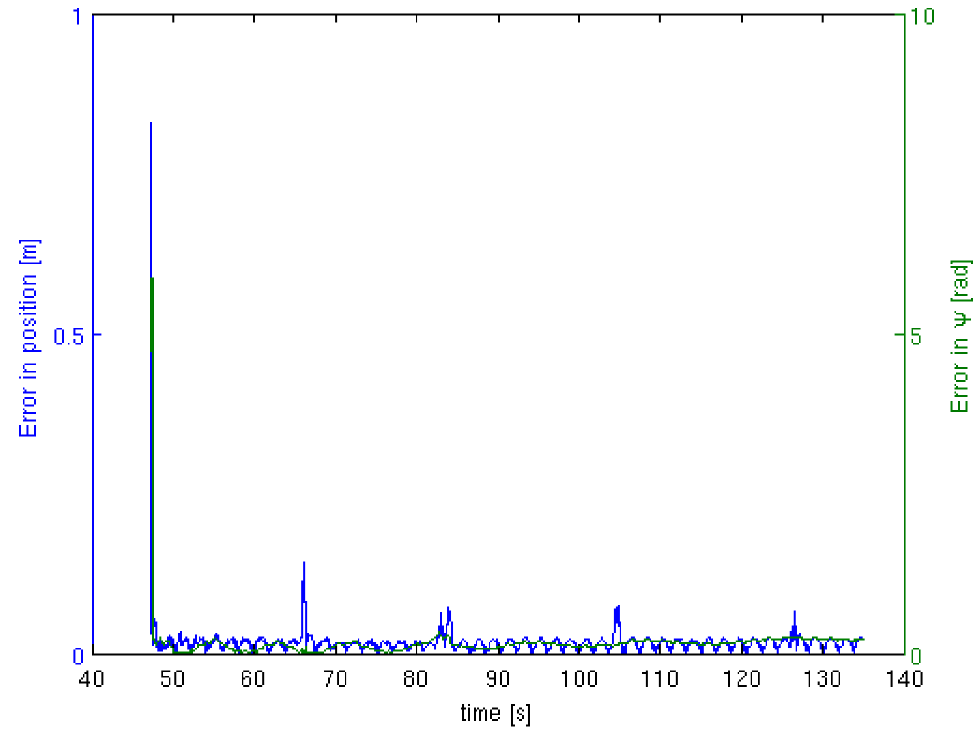


Figure 4.5: Error in position and orientation using a GPS signal with a SNR of 5 dB.

Ratios	$\text{meanError}_{pos}^{5\ dB}$	$\text{meanError}_{pos}^{25\ dB}$	$\text{meanError}_{pos}^{noGPS\ MTi-G}$	$\text{meanError}_{pos}^{noG}$
$\text{meanError}_{pos}^{5\ dB}$	1			
$\text{meanError}_{pos}^{25\ dB}$		1		
$\text{meanError}_{pos}^{noGPS\ MTi-G}$			1	
$\text{meanError}_{pos}^{noGPS\ PX4}$				1

Table 4.1: ratio pos.

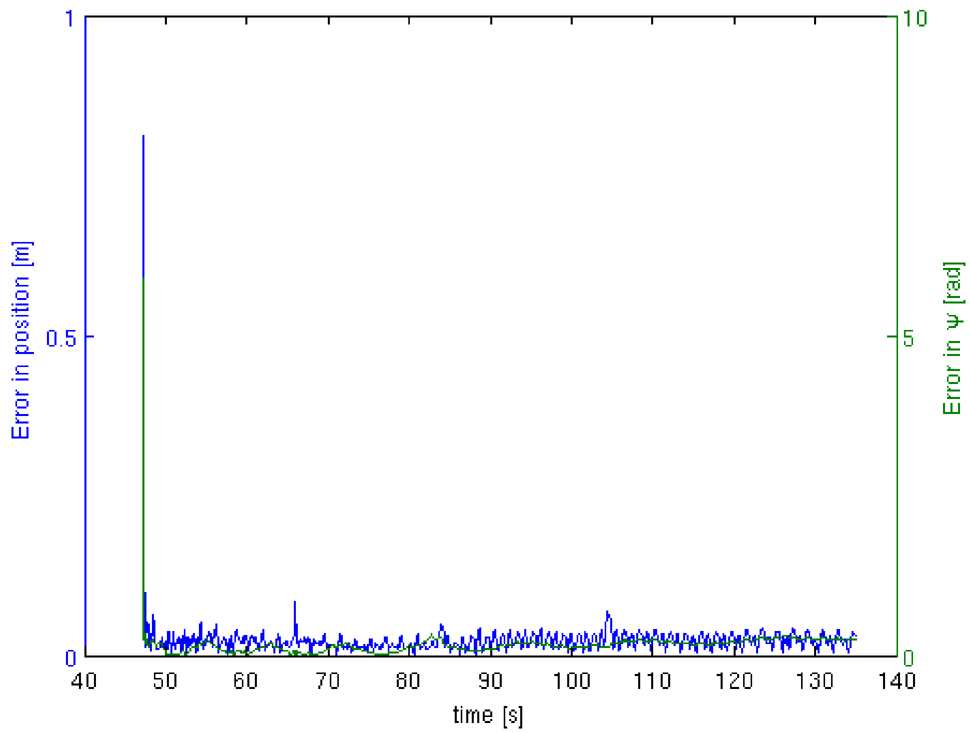


Figure 4.6: Error in position and orientation using a GPS signal with a SNR of 25 dB.

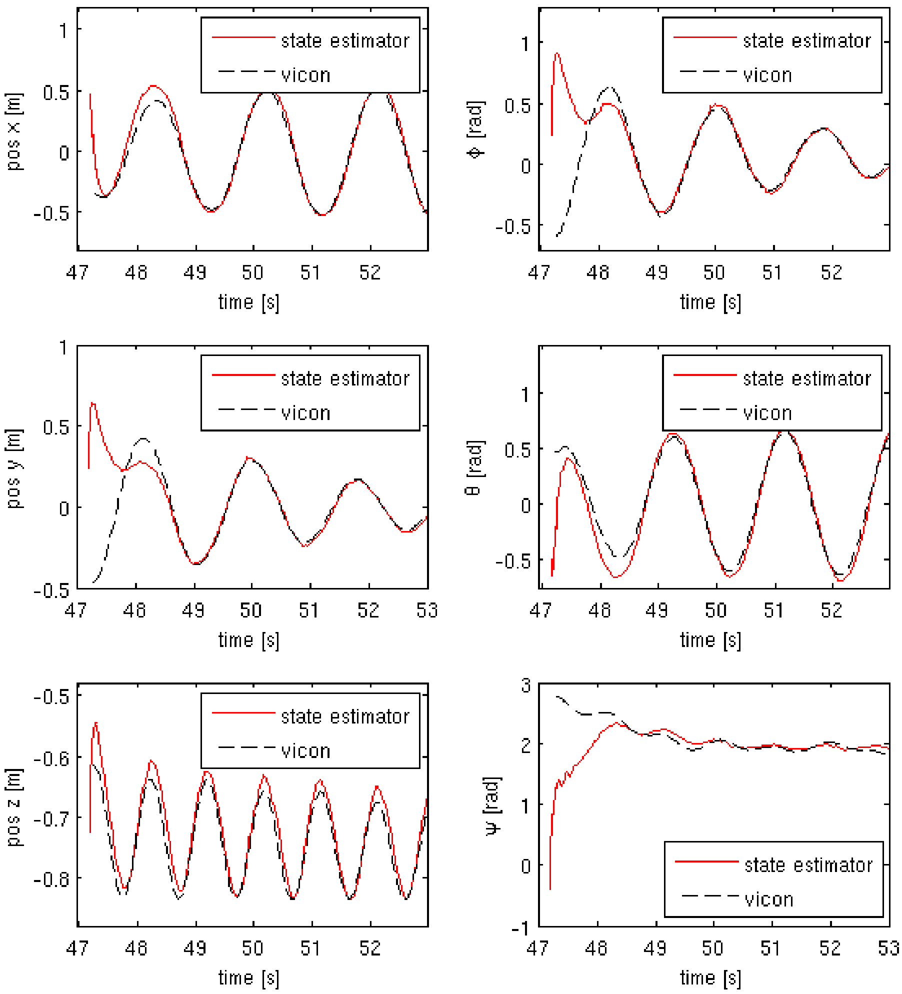


Figure 4.7: Estimation of the states without GPS signal.

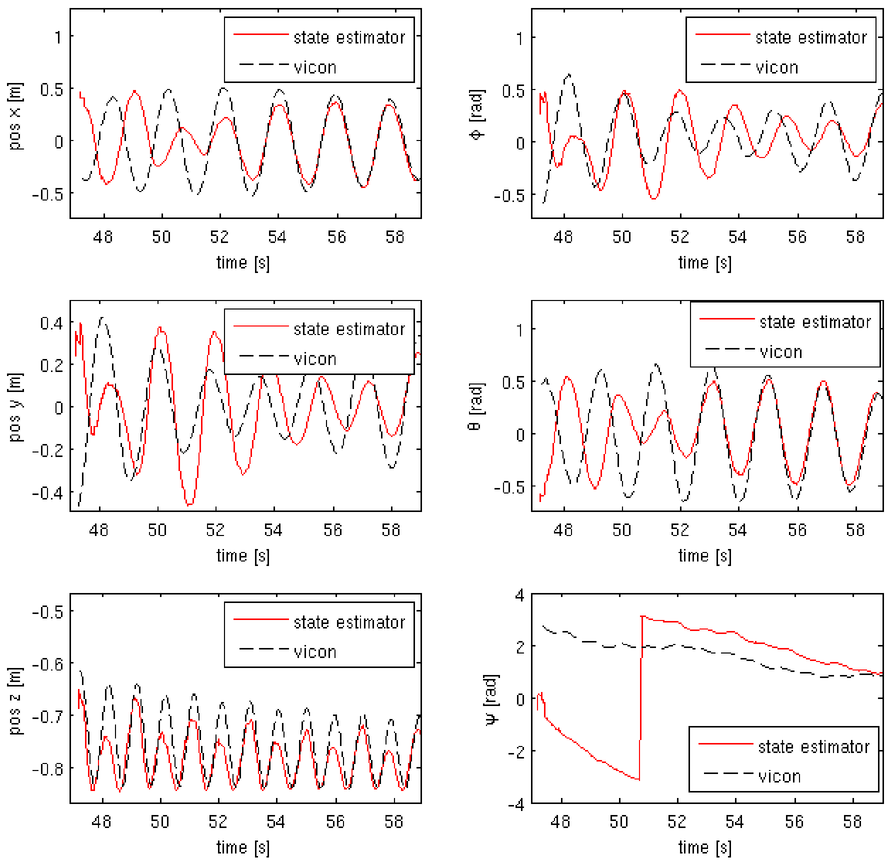


Figure 4.8: Estimation of the states without GPS and the help of the PX4 IMU.

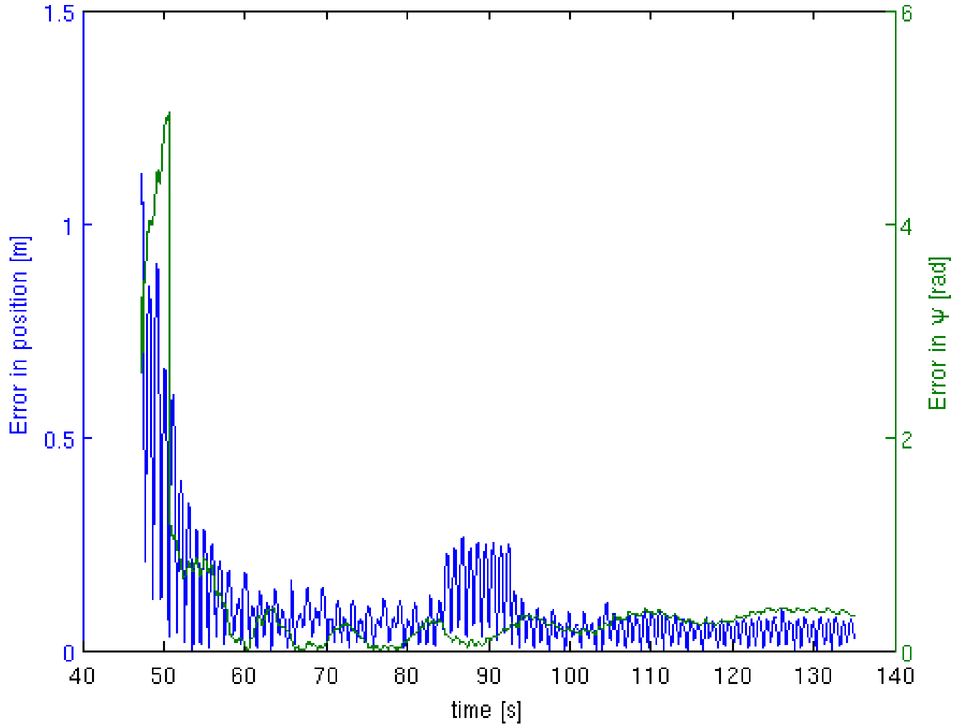


Figure 4.9: Error of the estimation of the states without GPS and the help of the PX4 IMU.

Chapter 5

Conclusion

Bibliography

[1] Miles L. Loyd. Crosswind kite power. *J. Energy*, 4(3):106–111, 1980.

[2] M. Canale, L. Fagiano, M. Ippolito, and M. Milanese. Control of tethered airfoils for a new class of wind energy generator. In *Decision and Control, 2006 45th IEEE Conference on*, pages 4020 –4026, dec. 2006.

[3] R. Schmehl. Large-scale power generation with kites. *J. Soc. Aerospace Engineering Students*, 1:21–22, 2012.

[4] M. Vukov S. Gros J. Swevers K. Geebelen, H. Ahmad and M. Diehl. An experimental test set-up for launch/recovery of an airborne wind energy (awe) system. In *In Proceedings of the 2012 American Control Conference, Montreal, Canada*, 2012.

[5] Michael Erhard and Hans Strauch. Control of towing kites for seagoing vessels. *CoRR*, abs/1202.3641, 2012.

[6] Swiss kite power. http://www.swisskitepower.ch. accessed 06-2012.

[7] András Héjj. Kalman-filter based position and attitude estimation algorithms for an inertial measurement unit.

[8] Ushanthan Jeyabalan. The ring on the hook.

[9] S. Sukkarieh, E.M. Nebot, and H.F. Durrant-Whyte. A high integrity imu/gps navigation loop for autonomous land vehicle applications. *Robotics and Automation, IEEE Transactions on*, 15(3):572 –578, jun 1999.

[10] Jong-Hyuk Kim, Salah Sukkarieh, and Stuart Wishart. Real-time navigation, guidance, and control of a uav using low-cost sensors. In ShinâĂŹichi Yuta, Hajima Asama, Erwin Prassler, Takashi Tsubouchi, and Sebastian Thrun, editors, *Field and Service Robotics*, volume 24 of *Springer Tracts in Advanced Robotics*, pages 299–309. Springer Berlin / Heidelberg, 2006. 10.1007/10991459$_2$9.

[11] D. Smith and S. Singh. Approaches to multisensor data fusion in target tracking: A survey. *Knowledge and Data Engineering, IEEE Transactions on*, 18(12):1696 –1710, dec. 2006.

[12] L. Debnath. *Nonlinear Partial Differential Equations for Scientists and Engineers.* Birkhäuser, 2005.

[13] W. Kutta. Beitrag zur näherungweisen Integration totaler Differentialgleichungen. *Z. Math. Phys.*, 46:435–453, 1901.

Appendix A

Data Sheets of the PX4 sensors

A.1 Data Sheet of the Accelerometer

BMA180
Digital, triaxial acceleration sensor

Bosch Sensortec

BOSCH
Invented for life

General description

The BMA180 is an ultra high performance digital triaxial acceleration sensor, aiming for low power consumer market applications.

The BMA180 allows high accurate measurement of accelerations in 3 perpendicular axes and thus senses tilt, motion, shock and vibration in cell phones, handhelds, computer peripherals, man-machine interfaces, virtual reality features and game controllers.

The BMA180 is highly configurable in order to give the designer full flexibility when integrating the sensor into his system. All features can be set by software via the digital interface. Here the user can choose between an I²C and an SPI (4-wire) interface mode.

BMA180 target applications

- Navigation (INS / Dead Reckoning)
- High accurate tilt detection (e.g. tilt compensated compass)
- Pointing and menu scrolling
- Display profile switching (portrait / landscape)
- Gaming
- Drop detection for warranty logging
- Shock detection
- Step-counting

Sensor operation

The BMA180 represents a new generation of digital acceleration sensors with a unique performance and feature set within 3 mm x 3 mm x 0.9 mm standard LGA package.

Key features BMA180
▶ All parameters and features user programmable
▶ 7 user programmable g-ranges and 10 selectable bandwidth settings
▶ Low-power consumption
▶ Ultra-low-power self-wake-up mode
▶ User programmable interrupt engine
▶ 0g offset regulation and in-field offset re-calibration with ultra-high accuracy
▶ SPI (4-wire) / I²C interface
▶ RoHS compliant, halogen-free

Technical data		BMA180
Sensitivity axes		x / y / z
Measurement range (switchable via SPI / I²C)		±1g, ±1.5g, ±2g, ±3g, ±4g, ±8g, ±16g
Sensitivity (calibrated)		1g: 8192LSB/g 1.5g: 5460LSB/g 2g: 4096LSB/g 3g: 2730LSB/g 4g: 2048LSB/g 8g: 1024LSB/g 16g: 512LSB/g
Resolution		14bit ⇨ 0.244mg (±2g range) (switchable 12 bit option)
Nonlinearity		±0.15% FS (±2g range)
Zero-g offset (ex-factory)		±15mg
Zero-g offset (after offset fine tuning)		±5mg
Zero-g offset temperature drift		±0.5mg/K
Noise density		150µg / √Hz
Bandwidth (switchable via SPI/I²C)	low pass	10Hz ... 1200Hz
	high pass	1Hz
	band pass	0.2 ... 300Hz
Digital input / output		SPI & I²C, interrupt pin
Supply voltages	V_{DD}	1.62 ... 3.6V
	V_{DDIO}	1.20 ... 3.6V
Current consumption		650µA
Temperature range		-40°C ... +85°C

A.2 Data Sheet of the Gyroscope

2 Mechanical and electrical specifications

2.1 Mechanical characteristics

@ Vdd = 3.0 V, T = 25 °C unless otherwise noted.

Table 3. Mechanical characteristics(1)

Symbol	Parameter	Test condition	Min.	Typ.(2)	Max.	Unit
FS	Measurement range	User-selectable		±250		dps
				±500		
				±2000		
So	Sensitivity	FS = 250 dps		8.75		mdps/digit
		FS = 500 dps		17.50		
		FS = 2000 dps		70		
SoDr	Sensitivity change vs. temperature	From -40 °C to +85 °C		±2		%
DVoff	Digital zero-rate level	FS = 250 dps		±10		dps
		FS = 500 dps		±15		
		FS = 2000 dps		±75		
OffDr	Zero-rate level change vs. temperature	FS = 250 dps		±0.03		dps/°C
		FS = 2000 dps		±0.04		dps/°C
NL	Non linearity	Best fit straight line		0.2		% FS
Rn	Rate noise density			0.03		$dps/(\sqrt{Hz})$
ODR	Digital output data rate			95/190/ 380/760		Hz
Top	Operating temperature range		-40		+85	°C

1. The product is factory calibrated at 3.0 V. The operational power supply range is specified in *Table 4*.
2. Typical specifications are not guaranteed.

A.3 Data Sheet of the Magnetometer

SPECIFICATIONS (* Tested at 25°C except stated otherwise.)

Characteristics	Conditions*	Min	Typ	Max	Units
Power Supply					
Supply Voltage	VDD Referenced to AGND	2.16	2.5	3.6	Volts
	VDDIO Referenced to DGND	1.71	1.8	VDD+0.1	Volts
Average Current Draw	Idle Mode	-	2	-	μA
	Measurement Mode (7.5 Hz ODR; No measurement average, MA1:MA0 = 00) VDD = 2.5V, VDDIO = 1.8V (Dual Supply) VDD = VDDIO = 2.5V (Single Supply)	-	100	-	μA
Performance					
Field Range	Full scale (FS)	-8		+8	gauss
Mag Dynamic Range	3-bit gain control	±1		±8	gauss
Sensitivity (Gain)	VDD=3.0V, GN=0 to 7, 12-bit ADC	230		1370	LSb/gauss
Digital Resolution	VDD=3.0V, GN=0 to 7, 1-LSb, 12-bit ADC	0.73		4.35	milli-gauss
Noise Floor (Field Resolution)	VDD=3.0V, GN=0, No measurement average, Standard Deviation 100 samples (See typical performance graphs below)		2		milli-gauss
Linearity	±2.0 gauss input range			0.1	±% FS
Hysteresis	±2.0 gauss input range		±25		ppm
Cross-Axis Sensitivity	Test Conditions: Cross field = 0.5 gauss, Happlied = ±3 gauss		±0.2%		%FS/gauss
Output Rate (ODR)	Continuous Measurment Mode	0.75		75	Hz
	Single Measurement Mode			160	Hz
Measurement Period	From receiving command to data ready		6		ms
Turn-on Time	Ready for I2C commands		200		μs
	Analog Circuit Ready for Measurements		50		ms
Gain Tolerance	All gain/dynamic range settings		±5		%
I²C Address	8-bit read address		0x3D		hex
	8-bit write address		0x3C		hex
I²C Rate	Controlled by I²C Master			400	kHz
I²C Hysteresis	Hysteresis of Schmitt trigger inputs on SCL and SDA - Fall (VDDIO=1.8V)		0.2*VDDIO		Volts
	Rise (VDDIO=1.8V)		0.8*VDDIO		Volts
Self Test	X & Y Axes		±1.16		gauss
	Z Axis		±1.08		
	X & Y & Z Axes (GN=5) Positive Bias	243		575	LSb
	X & Y & Z Axes (GN=5) Negative Bias	-575		-243	
Sensitivity Tempco	T_A = -40 to 125°C, Uncompensated Output		-0.3		%/°C
General					
ESD Voltage	Human Body Model (all pins)			2000	Volts
	Charged Device Model (all pins)			750	
Operating Temperature	Ambient	-30		85	°C
Storage Temperature	Ambient, unbiased	-40		125	°C

Appendix B

Data Sheets of the x-IMU sensors

B.1 Data Sheet of the Gyroscope

3 Electrical Characteristics

3.1 Sensor Specifications

Typical Operating Circuit of Section 4.2, VDD = 2.5V, VLOGIC = 2.5V, T_A=25°C.

Parameter	Conditions	Min	Typical	Max	Unit	Notes
GYRO SENSITIVITY						
Full-Scale Range	FS_SEL=0		±250		°/s	4
	FS_SEL=1		±500			4
	FS_SEL=2		±1000			4
	FS_SEL=3		±2000			4
Gyro ADC Word Length			16		Bits	3
Sensitivity Scale Factor	FS_SEL=0		131		LSB/(°/s)	1
	FS_SEL=1		65.5			3
	FS_SEL=2		32.8			3
	FS_SEL=3		16.4			3
Sensitivity Scale Factor Tolerance	25°C	-3		+3	%	1
Sensitivity Scale Factor Variation Over Temperature			±2		%	7
Nonlinearity	Best fit straight line; 25°C		0.2		%	6
Cross-Axis Sensitivity			2		%	6
GYRO ZERO-RATE OUTPUT (ZRO)						
Initial ZRO Tolerance	25°C		±20		°/s	1
ZRO Variation Over Temperature	-40°C to +85°C		±0.1		°/s/°C	7
Power-Supply Sensitivity (1-10Hz)	Sine wave, 100mVpp; VDD=2.2V		0.2		°/s	5
Power-Supply Sensitivity (10 - 250Hz)	Sine wave, 100mVpp; VDD=2.2V		0.2		°/s	5
Power-Supply Sensitivity (250Hz - 100kHz)	Sine wave, 100mVpp; VDD=2.2V		4		°/s	5
Linear Acceleration Sensitivity	Static		0.1		°/s/*g*	6
GYRO NOISE PERFORMANCE	**FS_SEL=0**					
Total RMS Noise	DLPFCFG=2 (100Hz)		0.1		°/s-rms	1
Rate Noise Spectral Density	At 10Hz		0.01		°/s/√Hz	3
GYRO MECHANICAL FREQUENCIES						
X-Axis		30	33	36	kHz	1
Y-Axis		27	30	33	kHz	1
Z-Axis		24	27	30	kHz	1
GYRO START-UP TIME	**DLPFCFG=0**					
ZRO Settling	to ±1°/s of Final		50		ms	5
TEMPERATURE SENSOR						
Range			-30 to 85		°C	2
Sensitivity	Untrimmed		280		LSB/°C	2
Room-Temperature Offset	35°C		-13200		LSB	1
Linearity	Best fit straight line (-30°C to +85°C)		±1		°C	2
TEMPERATURE RANGE						
Specified Temperature Range		-40		85	°C	2

Notes:

1. Tested in production
2. Based on characterization of 30 parts over temperature on evaluation board or in socket
3. Based on design, through modeling and simulation across PVT
4. Typical. Randomly selected part measured at room temperature on evaluation board or in socket
5. Based on characterization of 5 parts over temperature
6. Tested on 5 parts at room temperature
7. Based on characterization of 48 parts on evaluation board or in socket

B.2 Data Sheet of the Accelerometer and Magnetometer

2 Module specifications

2.1 Mechanical characteristics

@ Vdd = 2.5 V, T = 25 °C unless otherwise noted[a]

Table 3. Mechanical characteristics

Symbol	Parameter	Test conditions	Min.	Typ.[1]	Max.	Unit
LA_FS	Linear acceleration measurement range[2]	FS bit set to 00		±2.0		g
		FS bit set to 01		±4.0		
		FS bit set to 11		±8.0		
M_FS	Magnetic measurement range	GN bits set to 001		±1.3		gauss
		GN bits set to 010		±1.9		
		GN bits set to 011		±2.5		
		GN bits set to 100		±4.0		
		GN bits set to 101		±4.7		
		GN bits set to 110		±5.6		
		GN bits set to 111		±8.1		
LA_So	Linear acceleration sensitivity	FS bit set to 00 12 bit representation	0.9	1	1.1	m*g*/digit
		FS bit set to 01 12 bit representation	1.8	2	2.2	
		FS bit set to 11 12 bit representation	3.5	3.9	4.3	
M_GN	Magnetic gain setting	GN bits set to 001 (X,Y)		1055		LSB/gauss
		GN bits set to 001 (Z)		950		
		GN bits set to 010 (X,Y)		795		
		GN bits set to 010 (Z)		710		
		GN bits set to 011 (X,Y)		635		
		GN bits set to 011 (Z)		570		
		GN bits set to 100 (X,Y)		430		
		GN bits set to 100 (Z)		385		
		GN bits set to 101 (X,Y)		375		
		GN bits set to 101 (Z)		335		
		GN bits set to 110 (X,Y)		320		
		GN bits set to 110 (Z)		285		
		GN bits set to 111[2] (X,Y)		230		
		GN bits set to 111[2] (Z)		205		

a. The product is factory calibrated at 2.5 V. The operational power supply range is from 2.5 V to 3.3 V.