Lehrstuhl für Raumfahrttechnik
Prof. Prof. h.c. Dr. Dr. h.c. Ulrich Walter

Technische Universität München

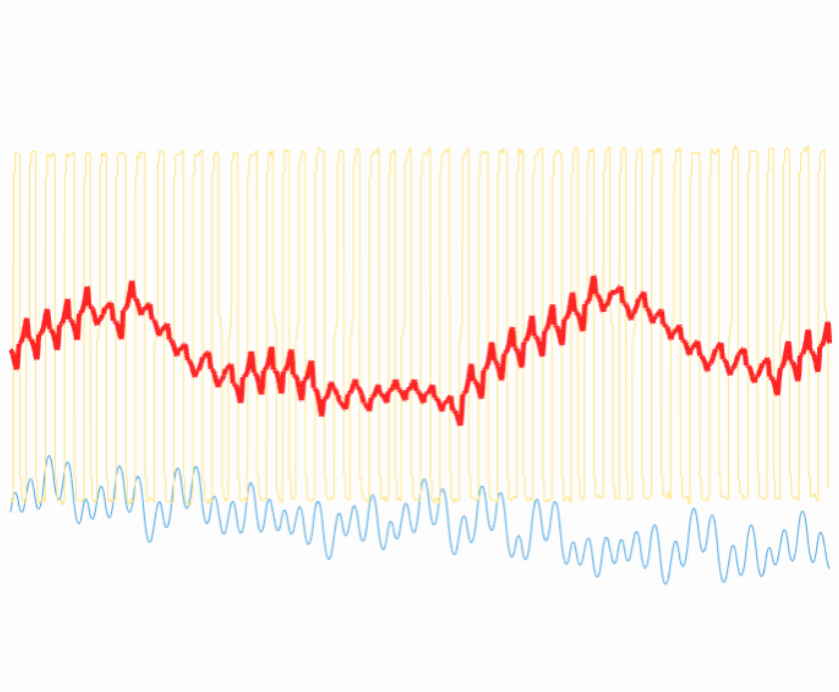
Semester Thesis

Design and Characterization of a Filter for MEMS Sensor Data for the Control of Cold Gas Actuators in Suborbital Missiles

RT-SA-2019/06

Author:

Flavio Rehn



Betreuer: Dipl.-Ing. (Univ.) Martin Dziura
Institute of Astronautics
Technische Universität München

Acknowledgments

I want to thank all project members of project Exosphere, the WARR e.V., the Institute Flight System Dynamics, and the Chair of Astronautics for contributing their technical knowledge to this work.

My special thanks go out to Till Assmann, Tim Klose, Saverio Nobbe and Matthias Beck for their continuous support.

Without Martin Dizura, who supervised this thesis, this work would not have been possible.

Zusammenfassung

Die WARR e.V. beteiligt sich am langfristigen Ziel, einen europäischen Kleinsatelliten-Startanbieter zu etablieren, indem sie ein Kaltgas-Lageregelungssystem (Englisch: reaction control system, RCS) entwickelt, um die Oberstufe einer suborbitalen Technologiedemonstrator-Rakete während einer instabilen Flugphase zu stabilisieren. Das RCS wird auf einer bodengebundenen Testanlage getestet, die einem inversen Pendel ähnelt. Diese Arbeit zielt darauf ab, die RCS-Entwicklung zu unterstützen, indem ein Filter zur Lagebestimmung mit kommerziell erhältlichen Komponenten (Englisch: commercial-of-the-shelf-components, COTS) entwickelt wird. Zwei Filteralgorithmen werden in einer Simulink-Umgebung entwickelt und getestet, welche aus mathematischen Modellen aller relevanten RCS-Subsysteme besteht. Einer der Algorithmen, basiert auf einem Komplementär Filter, während der andere auf einem adaptiven Kalman Filter basiert. Um den Tuning- und Implementierungsprozess der Filter auf dem Mikrocontroller zu erleichtern und die Grenzen des RCSs aufzuzeigen, werden die Filteralgorithmen durch Methoden der Statistischen Versuchsplanung in einem Monte-Carlo-Simulationsexperiments charakterisiert. Durch Kategorisierung und Verwendung der quadratischen Sigma-normierten Empfindlichkeit werden für jeden Filter die wesentlichen leistungsbeeinträchtigenden Faktoren ermittelt. Abhängig von der Filterkonfiguration zeigen sich die Latenz und das Gyroskoprauschen als besonders schädlich für die RCS-Performanz, welche durch die Wurzel aus dem gemittelten Fehlerquadrat der wahren Trajektorie zur vorgegebenen Trajektorie bestimmt wird. Insgesamt erfüllen beide Filter die an sie gestellten funktionalen Anforderungen. Wobei die Kalman Filter-basierte Variante eine bessere mittlere Performanz und die Komplementär Filter-basierte Variante eine höhere Robustheit bietet.

Abstract

The WARR e.V. participates in the long-term endeavour to establish a European small-satellite launch provider by developing a cold gas reaction control system (RCS) for stabilizing the upper stage of a suborbital technology demonstrator during instable flight. The RCS is tested on a ground-based test facility resembling an inverse pendulum. This thesis aims to support the RCS development by designing a filter for attitude determination with commercial-of-the-shelf-components (COTS). Two filter algorithms are developed and tested in a simulation-only Simulink environment consisting of mathematical models of all relevant RCS subsystems. The first will be based on a complementary filter, while the other is based on an innovation-based adaptive Kalman filter. To facilitate the tuning and implementation process of the filters on the microcontroller and to reveal the limitations of the RCS, the filter algorithms will be characterized by conducting a designed Monte Carlo simulation experiment. Through categorization and the use of the squared sigma-normalized sensitivity, performance impairing factors will be determined for each filter. Depending on the filter configuration, the computational delay and the gyroscope noise are revealed as especially detrimental to the RCS performance, which is measured as the root-mean-square error of the true trajectory to the commanded trajectory. Overall, both filters meet the functional requirements placed on them. However, the Kalman filter-based configuration provides a superior mean performance, while the complementary-based configuration provides a higher robustness.

Table of Contents

List of Figures

List of Tables

Symbols

Formula Symbols

Symbol	Unit	Description
σ	$[\sim]$	one standard deviation; $68.27\ \%$ -error
2σ	$[\sim]$	two standard deviations; $95.45\ \%$ -error
3σ	$[\sim]$	three standard deviations; $99.73\ \%$ -error
σ^2	$[\sim]$	variance
F_{diff}	$[\mathrm{N}]$	total thrust of 1-DoF RCS
K_C	$[-]$	Complementary filter gain
K_K	$[-]$	Kalman filter gain
P_0	$[\sim]$	initial error variance
S_x	$[-]$	derivative-based sensitivity towards x
S_x^σ	$[-]$	sigma-normalized sensitivity towards x
m_{gc}	$[\mathrm{kg}]$	gas consumption
s_X	$[-]$	switch to de / activate subsystem X within the model
$t_{PIDdelay}$	$[\mathrm{s}]$	microcontroller computational delay
$t_{actdelay}$	$[\mathrm{s}]$	actuator delay
t_{sim}	$[\mathrm{s}]$	simulation time
t_{st}	$[\mathrm{s}]$	actuator switching time
y_k	$[\sim]$	measurement innovation
θ_{ES}	$[°]$	end stop angle
ω_0	$[\mathrm{Hz}]$	natural frequency
$G(s)$	$[-]$	transfer function (frequency domain)
L	$[-]$	experiment sample size or levels in designed experiments
MAE	$[\sim]$	mean-absolute-error
N	$[-]$	number of samples / runs in a simulation
P	$[\sim]$	error variance
Q	$[\sim]$	system variance
R	$[\sim]$	measurement variance
$RMSE$	$[\sim]$	root-mean-square-error
a	$[\mathrm{m/s^2}]$	acceleration
$c(x)$	$[-]$	cummulative distribution function of x

$f(x)$	$[-]$	probability distribution function of x
$p(x)$	$[-]$	probability of x occurring
u	$[-]$	controller output
w	$[\sim]$	additive white gaussian noise on measurement
$\boldsymbol{M}$	$[-]$	linear scale matrix
ζ	$[-]$	relative damping
θ	$[°]$	true trajectory
μ	$[\sim]$	mean
ω	$[°/\mathrm{s}]$	angular velocity

Constants

Symbol	Value and Unit	Description
g	$9.807\ [\mathrm{m/s^2}]$	Gravitational acceleration
π	$3.142\ [-]$	Ratio of a circle's circumference to its diameter

Subscripts and Superscripts

Symbol	Description
A	accelerometer
AS	acceleration sensitivity
B	bias
CM	center of mass
G	gyroscope
QE	quantization error
RB	residual bias
TS	temperature sensitivity
$diff$	difference
est	estimated (only used in code)
$ex(d)$	(distorted) experimental data

fit	numerically fitted function
max	maximum
$meas$	measured
min	minimum
st	switching time
st	switching time
tot	total
$true$	refers to the true trajectory (this subscript is sometimes omitted)
σ	sigma-normalized

Subscript Order

$x_{k,(type),(Source),(coordinate\ axes)}$

$a_{k,meas,A,x}$	Acceleration measured by the accelerometer in the direction of the x-axes in the Sensor frame at the current time step

Mathematical Notation

Symbol	**Description**
$\boldsymbol{x}$	Correspond to a vector or a matrix
x	Corresponds to a scalar
$\hat{x}$	Corresponds to the value x estimated by a filter / estimator
$\dot{x}$	Corresponds to the time derivative of x
$\|x\|$	Corresponds to the absolute value of x
$x_{k\|k-1}$	Expresses x_k at a time step k based on a previous value x_{k-1}
$\partial f(x)/\partial x$	Expresses the partial derivative of a function $f(x)$ with respect to x
Δx	Expresses a change in x
$x!$	Corresponds to the factorial function of x
$\boldsymbol{I}$	Always corresponds to the Unity Matrix

Abbreviations

“x“D	x dimensional
AKF	adaptive Kalman filter
BB	bang-bang
CDF	cumulative density function
CF	Complementary filter
CFRP	carbon fiber reinforced polymer
COTS	commercial of-the-shelf
DOE	Design of Experiments
DoF	degree(s) of freedom
EKF	extended Kalman filter
Ex-4	Experimental Missile 4
GAE	geometric average error
GNSS	global navigation satellite system
hot	higher order terms
IAE	innovation-based adaptive estimation
IMU	inertial measurement unit
INS	inertial navigation system
KF	Kalman filter
LRT	Lehrstuhl für Raumfahrttechnik
MAE	mean-absolute-error
MEMS	micro-electro-mechanical systems
OFAT	one-factor-at-a-time
PDF	probability density function
PDS	partial derivative sensitivity
PID	proportional-integral-derivative
PWM	pulse width modulation
RCS	Reaction Control System
RMSE	root-mean-square-error
SIL	software-in-the-loop
SNS	sigma-normalized partial derivative sensitivity
STD	standard deviation
UKF	unscented Kalman filter
WARR	Wissenschaftliche Arbeitsgemeinschaft für Raketentechnik und Raumfahrt

1 Motivation

Every rocket needs a payload. The main purpose of a rocket is to transport this payload from point A to point B. This task becomes extremely costly if point B happens to be in space. Although a payload in space may generate an inconceivable amount of benefit, private companies, universities, and economically weak nations often lack the financial means to afford a “ticket” to space. However, recent development in the privatization of space, with companies like SpaceX and Blue Origin at the forefront, promises to advance space technologies at an unprecedented rate, vastly lowering the cost of space transportation. This promise “piques” the interest of an increasing number of companies and universities wanting to find both market and research potential in an orbit around Earth. NewSpace – as this advent of modern spaceflight is referred to by the media [1] – is represented by Fig. 1-1 which indicates a growing interest in the small satellite market.

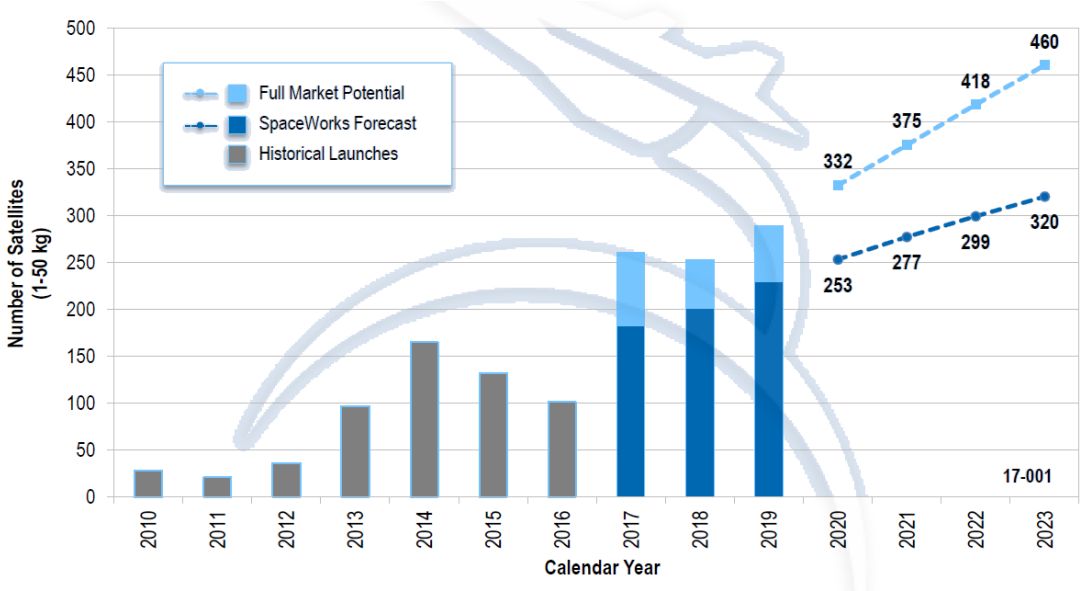


Fig. 1-1: Market forecast by SpaceWorks Enterprise Inc.®. Projections based on future plans of developers and programs indicate nearly 2400 nano / microsatellites will require a launch from 2017 through 2023 [2].

A significant part of projects involving small satellites stem from universities and private companies [3], which emphasizes the need for a cost-effective, customized, launch responsive and placement flexible space transport system. This demand can no longer be met by “piggyback” missions on traditional medium-to-heavy-lift launchers and is sparking the development of so-called microlaunchers: a technology that ignites the interest of a generation of space enthusiasts such as those within the Scientific Workgroup for Rocketry and Spaceflight (WARR) at the Technical University of Munich.

1.1 Introducing WARR Exosphere

The student group WARR e.V. (German: Wissenschaftliche Arbeitsgemeinschaft für Raketentechnik) began in 2017 with theoretical work on a microlauncher, which is an orbital launcher with a payload of a few kilograms. Comparable launchers have already

been developed such as the “Electron” rocket from the company Rocket Lab and the Japanese SS-520 [4]. Until spring 2018, the first two microlauncher workshops produced two carrier concepts with payloads ranging from 25 to 50 kilograms. In order to obtain the necessary funding for the realization of such a project, WARR decided to develop a suborbital technology demonstrator for testing all necessary technologies for an orbital carrier. The project called "Exosphere" was launched to develop the two-stage suborbital rocket dubbed WARR Experimental Missile 4 (WARR Ex-4). The development of the WARR Ex-4 should be as cost-effective as possible in the shortest possible time. Therefore, commercial-of-the-shelf (COTS) components shall be used. Some important components, which are necessary for the technology demonstration of an orbital rocket and not available on the market, are the stage adapter for performing the separation of the two stages, the recovery system for high re-entry speeds, and the active attitude control system. The last of which is also referred to as a reaction control system (RCS) and has the task of controlling and stabilizing the upper stage of the WARR Ex-4 along the vertical axes during a long thrustless flight phase. This shall allow the second stage thrusters to ignite at an optimal attitude. The development of the RCS was the primary focus of WARR’s third microlauncher workshop and will be the future focus of project Exosphere endeavors. All planned project phases, including the development of a test stand and precursor tests on the WARR EX-ONE, are detailed in Section 3.1.

1.2 Significance of this Work in the Project

As a part of project Exosphere, this thesis aims to support the development of the RCS. To control the attitude of any object, first, its attitude needs to be determined as accurately as possible. This is achieved through a multitude of sensors. However, COTS sensors provide faulty measurements to the control algorithm, degrading the accuracy and stability of the RCS. Therefore, the measurements of the different sensors are filtered to extract the optimal estimation of the current attitude of the system. This process will be referred to as attitude estimation based on inertial sensors. Applying this concept to the cold-gas RCS and identifying performance degrading factors will be the aim of this thesis. For this purpose, the current advances in microlauncher RCSs, COTS sensors, and filter algorithms will be examined in the sections 2.1-2.3. Following this, Chapter 3 will detail the objective of this thesis in the form of three research questions, place it in the overall context of the project, and elaborated on the concept behind testing the RCS. The main body of this work will explain the implementation of a plant model, a preliminary control algorithm (see Chapter 4), and multiple sensor data filtering algorithms (see Chapter 5) in a simulation environment. A designed Monte Carlo simulation experiment will be performed, evaluated, and discussed to characterize the filter throughout both Chapter 6 and 7. To conclude this work, a summary and the project outlook is given in Chapter 8.

Additionally, a guide on how to use the Simulink model of the RCS, including all mentioned subsystems, is appended in C.

2 State of the Art

The following sections give an overview of the state-of-the-art technologies crucial for attitude determination and control on small-scale suborbital rockets. In the project it was decided to settle on an RCS with cold gas thrusters over other more complex technologies. A justification for this as well as an extensive list of viable attitude control technologies on microlaunchers is provided in the semester thesis of fellow team member Till Assmann [5], who worked on the RCS hardware design.

2.1 Cold Gas Reaction Control Systems on Suborbital Missiles

In cold gas systems, a gas is stored in a tank at a sufficiently high pressure and expanded through a nozzle. Inert gases such as nitrogen, helium, krypton, argon, but also hydrogen and methane are often utilised [6]. However, it is also possible to use gases which are present in liquid form at high pressure. These include carbon dioxide, butane, propane, dinitrogen monoxide and ammonia among others. However, this often leads to the same problems as with liquid engines such as fuel swabbing and the need to heat the pipes and tanks, so that the gas remains gaseous [7]. An example of a suborbital rocket using an RCS with cold gas thrusters is the FALCO-4 model rocket.

The FALCO-4 was developed for vertical stabilization experiments on a model rocket at low speeds. It used $0.068\ \mathrm{kg}$ of carbon dioxide as a propellant, which was stored at $60\ \mathrm{bar}$ in a gas cartridge [8]. A pressure regulator expands the gas to $6\ \mathrm{bar}$. The system has four valves, each with one nozzle and none of these are Laval nozzles. The gas flows out directly from the open end of the gas line after the valve. The four nozzles are installed with an offset to the longitudinal axis of the rocket. Each nozzle can achieve a maximum thrust of $0.37\ \mathrm{N}$ [8]. The RCS data is compiled in Table 2-1.

Table 2-1: FALCO-4 RCS specifications from [8].

Parameter	**Value**
Overall RCS mass	$0.834\ [\mathrm{kg}]$
Propellant mass	$0.068\ [\mathrm{kg}]$
Thrust	$0.32 - 0.37\ [\mathrm{N}]$
Nozzle chamber pressure	$6\ [\mathrm{bar}]$
Tank pressure	$60\ [\mathrm{bar}]$
Number of nozzles	$4\ [-]$

Another example of a suborbital rocket using a cold gas RCS is the SHEFEX-II developed by the German Aerospace Center (German: Deutsches Zentrum für Luft- und Raumfahrt, DLR). In total, the whole control system includes three colds gas RCSs. One for precision control before ignition of the second stage, one for roll rate control during precision control, and one triaxial RCS for alignment of the upper stage after burnout for position update, alignment in the re-entry direction, and rotational rate damping during re-entry. For more information on this mission refer to [9].

Orbital rockets like the Falcon 9 by SpaceX use nitrogen fuelled cold gas RCSs for attitude control on the upper stage during the phase without thrust, and for attitude control on the lower stage during landing [10].

Advantages of cold gas RCSs are low complexity, relative security and reliability, low cost, the possibility to use non-toxic fuels and the ability to work in pulsed mode. However, only a relatively low thrust can be produced per nozzle [6].

2.2 Micro-Electro-Mechanical Systems Inertial Measurement Unit

Micro-electro-mechanical systems (MEMS) is a process technology used to create tiny integrated devices or systems that combine mechanical and electrical components. These devices are fabricated using integrated circuit batch processing techniques and can range in size from a few micrometres to millimetres. These systems have the ability to control, actuate and sense on the microscale [11]. The latter will be of importance for the inertial measurement unit and hence for all upcoming sections.

Inertial sensors comprise accelerometers and gyroscopes. An accelerometer measures a specific force and a gyroscope measures an angular rate, both without an external reference. Devices that measure the velocity, acceleration, or angular rate of a body with respect to features in the environment are not considered inertial sensors [12]. Most types of accelerometer measure the specific force along a single sensitive axis. Similarly, most types of gyroscopes measure the angular rate about a single axis. An inertial measurement unit (IMU) combines multiple accelerometers and gyros, usually three of each, to produce a three-dimensional measurement of specific force and angular rate [12].

MEMS IMUs have a significant cost, power consumption, mass and size advantage over conventional mechanical or optical (only on gyroscopes) inertial sensor designs. However, most MEMS sensors suffer from relatively poor performance on bias instability, scale factor instability and noise, which emphasizes the importance of extensive calibration [12, 13]. The performance degrades even further for the lowest grade of inertial sensors – the consumer grade. The popular MPU-6050 [14] is an example of a low-end consumer grade IMU primarily used in smartphones. Being mass produced, it impresses with high availability and cost efficiency, which qualifies it as a COTS component for the RCS. It incorporates a 3-axis MEMS gyroscope and accelerometer on the same silicon die. The following sections explain the working principle of two inertial sensors.

2.2.1 Accelerometers

MEMS accelerometers are commonly realized as pendulous accelerometers or vibrating beam accelerometers [12]. In both cases, a proof mass is free to move with respect to the casing along the sensitive axis. In the MPU-6050, a closed-loop force-feedback pendulous accelerometer is integrated [15]. This will be explained first.

2.2.1.1 Pendulous Accelerometers

In a force-feedback accelerometer, an electromagnet mounted on a pair of permanent magnets of opposite polarity, mounted on either side of the casing, is used to maintain the pendulous arm at a constant position with respect to the casing. The capacitive or

resistive pick-off detects departures from the equilibrium position. The force exerted by the electromagnet, which keeps the pendulous at the equilibrium point and parallel to the casing, is measured [12]. Fig. 2-1 depicts a mechanical force-feedback pendulous accelerometer.

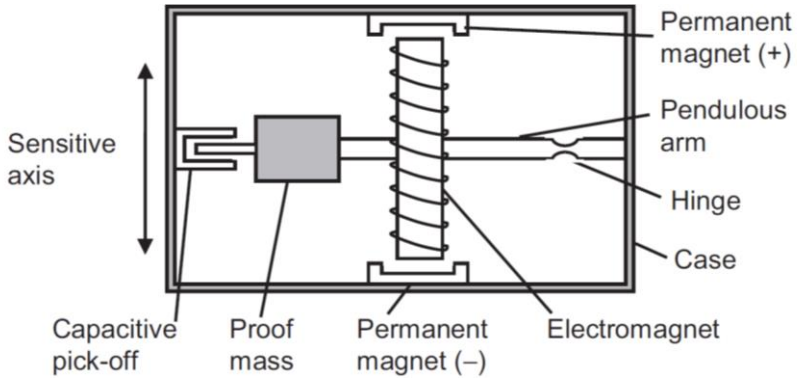


Fig. 2-1: Sketch of a mechanical closed-loop force-feedback pendulous accelerometer from [12].

The closed-loop configuration ensures that the sensitive axis remains aligned with the accelerometer casing, while the electromagnetic torquer offers much greater range and linearity than the open-loop accelerometer, which consists of a spring and a pick-off [12].

2.2.1.2 Vibrating-Beam Accelerometers

Another possible way of designing a MEMS accelerometer is the so-called vibrating-beam accelerometer. In this configuration, the proof mass is supported along the sensitive axis by a vibrating beam, largely constraining its motion with respect to the casing. When a force is applied to the accelerometer casing along the sensitive axis, the beam is compressed or stretched while pushing or pulling the proof mass. The beam is driven to vibrate at its resonant frequency by the accelerometer electronics. As the beam is compressed, the resonant frequency is decreased, while tensing it will cause the opposite. Now, the specific force along the sensitive axis can be determined by measuring the modified resonant frequency by [16]. Vibrating-beam accelerometers are considered an open-loop device; however, the proof mass is essentially fixed. Therefore, no variation in the sensitive axis with respect to the casing occurs.

Fig. 2-2 outlines the operating principle of the vibrating-beam accelerometer and Fig. 2-3 shows the concept of high-resolution quartz vibrating-beam accelerometer developed by [16]. In Fig. 2-2, F_0 represents the resonant frequency, while F represents the modified measured frequency. The acceleration denoted with Γ points in the direction of the sensitive axis. This notation is exclusive to this section and will not be used again.

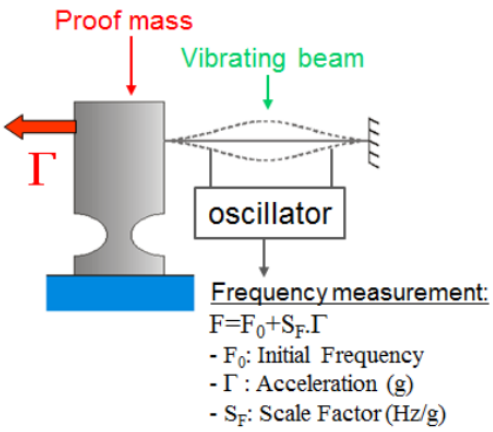


Fig. 2-2: Principle of operation of the vibrating-beam accelerometer taken from [16].

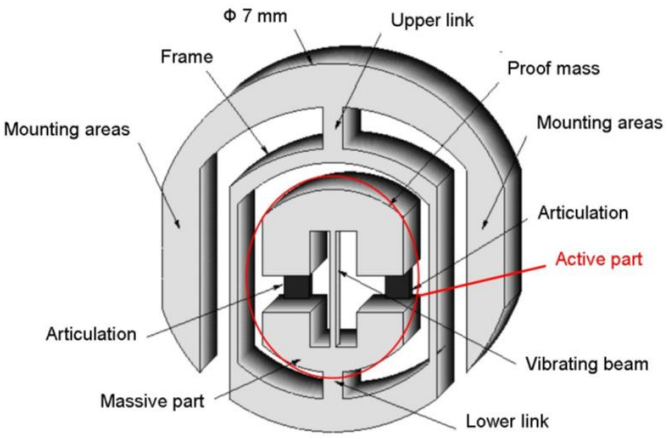


Fig. 2-3: Concept of a $50\,\mathrm{ng}$ quartz vibrating-beam inertial accelerometer by [16].

Such MEMS accelerometer designs have a typical total operating range of $\boldsymbol{a}_{range} = 4\,\mathrm{g}$. This translates to a minimum measurement of $-2\,\mathrm{g}$ and a maximum measurement of $2\,\mathrm{g}$. By increasing the operating range of such a device, the quantization error is increased as well. For traditional mechanical accelerometer designs, the operating range can be $200\,\mathrm{g}$ in total [12]. The quantization error and other error sources are explained in Section 4.5.

Furthermore, novel graphene-based accelerometer designs, making use of optical and MEMS technology, promise to combine higher accuracy with the advantages of MEMS [17]. However, due to its novelty, the technology cannot be considered a COTS component.

2.2.2 Gyroscopes

All MEMS gyroscopes operate on the vibratory principle [12]. Therefore, optical gyroscope designs like the ring laser gyroscope or the interferometric fiber-optic gyroscope will not be explained within this section.

A vibratory gyroscope comprises an element that is driven to vibrate at its natural frequency. The vibrating element may be a string, beam, ring or hemisphere among other structures. Fig. 2-4 illustrates a string mounted on a rectangular support or mount which vibrates about the centre of the gyroscope frame. The string tends to vibrate in the plane spanned by the drive axis and the input axis, even if its support rotates. The Coriolis acceleration instigates a harmonic motion along the axis perpendicular to both the driven vibration and the angular rate vector input axis. The amplitude of this motion is proportional to the angular rate ω around the input axis. A rotation about the drive / vibration axis does not produce a Coriolis acceleration. Therefore, only rotation about the input axis leads to significant oscillation in the output axis [12].

Fig. 2-5 shows a gyroscope design with a ring as a vibratory structure, with the input axis still being perpendicular to the drive and output axis. Here, four drive units are positioned at right angles around the input axis and four detectors are placed at intermediate points. When the gyro is not rotating, the detectors are at the nodes of the vibration mode and therefore, no signal is detected [12]. When an angular rate, ω,

is applied, the vibration mode is rotated about the input axis, allowing the pick-offs to measure an amplitude proportional to the input.

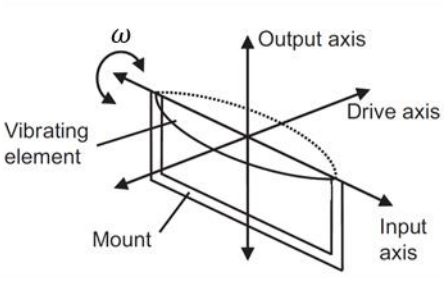


Fig. 2-4: Axes of a vibratory gyroscope with a string as the vibrating element [12].

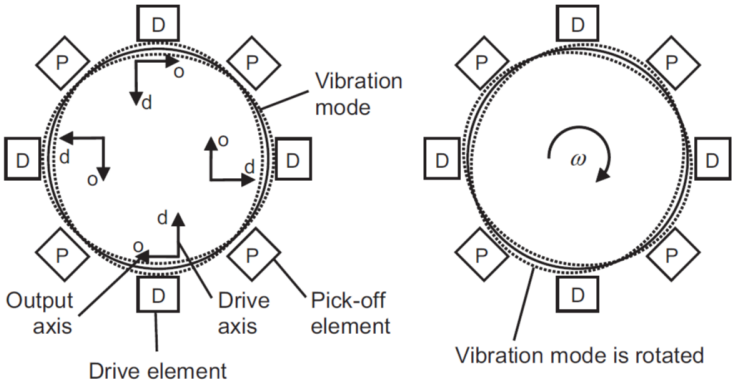


Fig. 2-5: Vibration modes of ring, cylinder and hemispherical vibratory gyroscope [12].

The vibratory gyroscope designs have a significant cost and power consumption advantage over the other mentioned designs. Operating ranges can reach from $\boldsymbol{\omega}_{range} = \pm 3\, rad/s$ to $\boldsymbol{\omega}_{range} = \pm 120\, rad/s$ [12].

Most MEMS gyroscope designs suffer from unreliability caused by unstable micro-mechanical structures. Further, vibrations and accelerations of high magnitude may cause additional errors, which are referred to as acceleration sensitivity [18]. Additionally, the complex electronic measuring circuitry increases the cost, with MEMS gyroscopes usually being more costly than MEMS accelerometers [13]. Therefore, novel chip-scale integrated optical gyroscopes are being researched which promise great potential in the low-cost consumer electronics market [18].

2.3 Attitude Estimation Based on Inertial Sensors

The measurements of the accelerometer and the gyroscope can both be used to determine the attitude of the rocket body. However, the accelerometer suffers from high measurement noise, while the gyroscope suffers from integration drift. To overcome these issues, an estimation algorithm is implemented on a microcontroller, which combines both measurement information. These estimation algorithms are referred to as filters. Combining the IMU with the filter algorithm and any global navigation satellite system (GNSS) receiver results in an inertial navigation system (INS). However, at this stage of the project, a GNSS module is not required. Therefore, the attitude determination relies solely on inertial sensors. A multitude of filter algorithms can be used for this task. However, more sophisticated estimation algorithms rely on high computing power, which is limited by the flight computer.

In commercial unmanned aerial vehicles (UAV), the complementary filter is often applied as an attitude estimation algorithm because of its simplicity and computational efficiency [19]. However the accuracy of such an algorithm is usually worse compared to the Kalman filter [20, 21]. The Kalman filter is a recursive estimator which theoretically gives the optimal estimation of a state vector of the true system by using a dynamic model and measurements, both corrupted with random noise of known statistics [22].

The Kalman filter comes in different “flavours” with the most important being the standard Kalman filter (KF), the extended Kalman filter (EKF) and the unscented Kalman filter (UKF). The EKF linearizes the nonlinear model equation about the current state estimate, while the standard KF does not [12]. This, however, requires the calculation of the Jacobian matrix, which requires higher computational effort [20]. The UKF is characterized by a set of sigma points to approximate a Gaussian probability distribution. The accuracy of the estimation for the UKF is expected to be higher than the EKF. However, the computational effort per iteration increases as well [12].

Going through the four mentioned filters, increasing accuracy seems to trade with increasing computational effort. Considering the limited computational power of the on-board computer, the Teensy 3.6 (see 3.1.1), first, the simple complementary filter and the standard Kalman filter shall be analysed. If, based on this, the project requirements in section 3.1.3 can be fulfilled, more sophisticated algorithms will not be implemented in the simulation. The theoretical background of all implemented filters is given in Chapter 5.

After giving an overview of a selection of fundamental technologies involved in the development of the RCS, the objective of this thesis is substantiated, and the concept of the RCS is described in the next chapter.

3 Objective of the Thesis

Introduced in 1.2, the general aim of this work is to design an appropriate filter for the stabilization and control of the second stage of a suborbital launch vehicle. Under the assumptions, which will be explained in 3.1, the second stage is simplified to 1-DoF inverse Pendulum at the current phase of the project. This results in the following research questions regarding the design (Q1) and the characterization (Q2 and Q3) of the filter:

Q1: Which filter design can **reliably** estimate the attitude of a **1-DoF inverse pendulum**, stabilized by a **cold gas reaction control system**, using the measurements of a **cost-effective IMU**, to control the system under the given **requirements and circumstances**?

In this sense, the reliability of the filter shall be defined as its ability to meet the performance requirements (see 3.1.3) while being confronted with the problems arising from choosing consumer grade IMUs (see 2.2), relatively high-g dynamics (see 4.4) and limited processing power of the on-board microprocessor (see Table 3-2).

Q2: What are the **key factors** influencing the control system?

Key factors shall be defined as a set of model input variables, which critically impair the performance of the system (see 6.1).

Q3: Which **rules** apply to **tuning** the control system?

Formulating the tuning rules shall address the questions: What are the limits of the RCS when varying key factors? Can tuning guidelines be formulated to support the tuning process of the filter? In this sense, characterizing the filter will be achieved through evaluating the whole control system performance.

Further, this document shall educate new and experienced fellow project members in the underlying mechanics of the RCS and the test stand by addressing the research questions and providing a comprehensible simulation environment (see 3.2 and B).

3.1 Scenario Description

Before Project Exosphere, there was the ambitious long-term goal to design and build a microlauncher, which can carry up to $50\ \mathrm{kg}$ of payload into a low earth orbit. To attract interest in a sustainable German launch provider and to gain technical experience in space bound multi-stage rockets, it was decided to test all necessary technology at a smaller scale. This technology demonstrator will be a two-staged suborbital rocket reaching space in a parabolic arc before falling back to earth just a few minutes later.

The WARR EX-4 suborbital rocket will be propelled by an already developed and tested $10\ \mathrm{kN}$ hybrid engine, for the lower stage, and a $2\ \mathrm{kN}$ liquid engine for the upper stage. The lower stage engine was developed by the WARR e.V. Rocketry team, while the upper stage engine is currently being developed by a Munich-based aerospace start-up. This leaves the structure, the staging system, the two-stage recovery system and the RCS to be developed within Project Exosphere. Due to limited man-power, it was decided to focus on the RCS which will actively stabilize the second stage after separation during a thrustless coasting phase. The stabilization is mandatory to keep

the rocket trajectory on a vertical path into space, eventually crossing the Kármán-line at $100\ \mathrm{km}$ height. The details of the WARR EX-4 mission are comprised in [5].

The development of the RCS was divided into six steps. The first steps comprise a reduction of complexity by introducing a single DoF to be stabilized on an inverse pendulum test stand. After scaling up to 3-DoF, the RCS will be tested and verified on the WARR EX-ONE model rocket. All six steps are comprised in the table below.

Table 3-1: The six phases for developing the RCS.

Phase	Goal
P0	The RCS shall regulate the pitch axis of an inverse pendulum test stand with external gas and power supply, while the yaw and roll axis are locked
P1	The RCS shall regulate the pitch and yaw axis of an inverse pendulum test stand with internal gas or power supply, while the roll axis is locked
P2	The RCS shall regulate the pitch, yaw and roll axis of an inverse pendulum test stand with internal gas and power supply
P3	The RCS shall be able to run a pre-programmed trajectory on the pendulum test stand with internal gas and power supply
P4	Development of RCS flight version and testing on the inverse pendulum
P5	Testing the RCS flight version on a modified WARR EX-ONE model rocket

No dedicated inverse pendulum test stand is available to the WARR and therefore must be designed as well. The following sections explain the concept behind the RCS and the inverse pendulum test stand.

3.1.1 The Reaction Control System Concept

The baseline of our attitude control system design is a pressure-fed cold-gas RCS, using nitrogen as a propellant, specifically tailored for its use during the thrustless costing phase [5]. Thrust vectoring and Vernier thrusters, where excluded as a design choice since they require the main engine to be active. The RCS was designed with scalability in mind. Therefore, the basic actuator design will remain unchanged for versions deployed on the test stand / WARR EX-ONE as well as those qualified for the final suborbital rocket. This work will focus on the first-mentioned ones, which are developed for the phases P0-P5.

The main hardware components of the RCS comprise: a carbon fiber reinforced polymer (CFRP) tank used for paintball guns; a compact, light-weight pressure regulator; a drain valve, fast solenoid nozzle valves; custom nozzles detailed in [5]; the electric power distribution system; the MPU-6050 IMU and the microcontroller to run the estimation and control algorithms. Fig. 3-1 illustrates the segments of the WARR EX-ONE “Lenk” version.

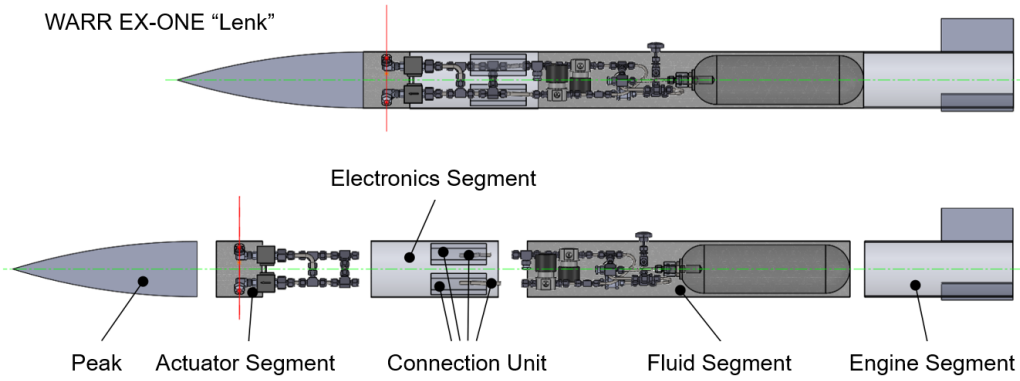


Fig. 3-1: WARR EX-ONE “Lenk” version, with a diameter of $0.12\ \mathrm{m}$, translated from [5].

Considering the main design goals, cost-efficiency, availability, size and weight, the hardware mostly consists of COTS components. Important to this work will be the actuator segment, illustrated in Fig. 3-2, which contains the nozzles and the solenoid valves. These elements will decide on the maximum thrust and transient behavior of the actuators which will have a major effect on the control system performance.

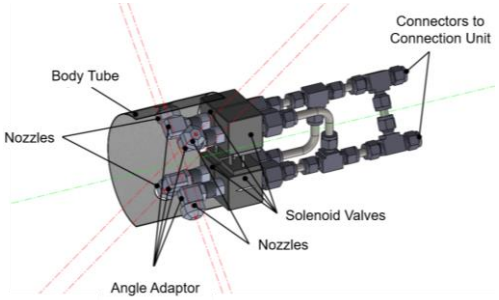


Fig. 3-2: Overview actuator segment of the WARR EX-ONE “Lenk” translated from [5].

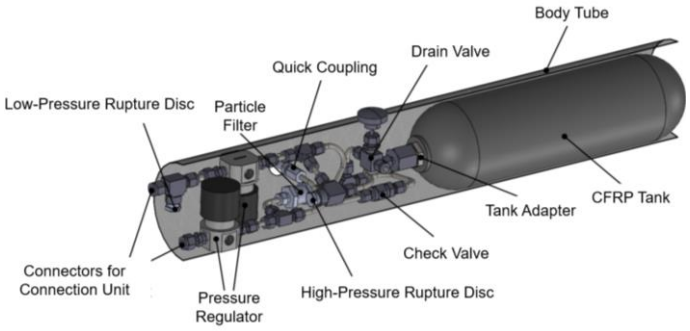


Fig. 3-3: Overview fluid segment of the WARR EX-ONE “Lenk” translated from [5].

Following the example of [8], a four nozzle configuration with offset to the center was chosen to control the 3D attitude. This is expected to save space, cost and weight. Hereby, the fast responding solenoid valves represent an especially costly component.

The fluid segment, the electronics segment and the actuator segment will be mounted on the test stand for P1 and onwards. However, at the point this work was composed, only two sufficiently fast acting solenoid valves were available to the WARR. For this reason, and to further reduce the complexity of the stabilization task, all following sections consider the 1-DoF case of P0 and P1. Therefore, the hardware design team calculates a maximum thrust of $F_{max} = 7\ \mathrm{N}$ per nozzle by using the available Festo MHE4 solenoid valves [23], assuming nitrogen as the propellant. An overview of all parameters needed for the Simulink model will be given in section 6.1 and throughout Chapter 4.

The electronics segment of the RCS will contain the power supply, wiring and circuitry, the on-board computer, the IMU and the communication module. The power supply and circuitry board are, at the point of this work, still being developed. The on-board computer was chosen to be a Teensy 3.6 (see Fig. 3-4). It is superior to the Arduino

UNO in processing speed while being more compact. It comes equipped with a I2C bus, which will be required when receiving the $16\ \mathrm{bit}$ measurements of the IMU at high frequencies on six inputs. The specifications of the Teensy 3.6 are summarized in the table below.

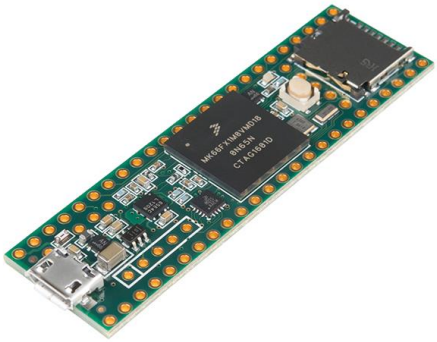


Fig. 3-4: Picture of the Teensy 3.6 taken from [24].

Table 3-2: Teensy 3.6 technical specification from [25].

Feature	**Teensy 3.6**
Processor	MK66FX1M0VMD18
Core	Cortex-M4
Rated Speed	180 [MHz]
Flash Memory	512 [kB]
RAM	256 [kB]
Voltage	3.3 [V]
EEPROM	4 [kB]
Interfaces	USB, CAN, I2C, SPI, Ethernet, SD, I2S

The MPU-6050 was chosen as the IMU, due to its low cost and availability. It will measure the acceleration, $\boldsymbol{a}$ in $\mathrm{m/s^2}$, and the angular rate, $\boldsymbol{\omega}$ in $°/\mathrm{s}$, along three axes. However, for P0 and P1 only the blue marked states will be filtered and fed to the control algorithm:

$$\boldsymbol{a} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix};\ \boldsymbol{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}. \qquad \textbf{Eq. (3–1)}$$

Now, the purpose of the RCS will be to control the angular deflection, θ, around the y-axis of the inertial frame of the test stand to zero. This purpose is further defined by the functional requirements in 3.1.3, after introducing the test stand design in the next section.

3.1.2 The Test Stand

The test stand was designed with the goal in mind to simulate the instable flight of the coasting phase for the EX-4 and the fin-stabilized flight of the EX-ONE. A simple modifiable inverse pendulum fulfils this requirement.

The EX-4 coasting phase is not aerodynamically stabilized and therefore assumes instable flight. Side winds will easily destabilize the rocket, which would diminish the achievable vertical distance. This scenario cannot be tested on a classical pendulum design, since it would not reproduce the behaviour of an instable-flying rocket as it is considered a stable system. This problem is solved by inverting the pendulum and thus positioning the RCS and together with the overall centre of mass above the pivot. In its upright position, the inverted pendulum occupies an unstable position, thus requiring an active control to remain upright. When the pendulum is deflected, the lever arm and

thereby the occurring torque is proportional to the sine function of its angle. This is a comparable response to a disturbance at the EX-4 coasting phase, making this design well suited for the given task.

For the second scenario on the EX-One, the stable flight conditions can be simulated by introducing a counterweight under the pivot to move the centre of mass below it. A mission on this configuration could comprise simply holding a $\theta = 5\,°$ deflection, with the help of the RCS, over a specified period.

The final design, constructed by the testing team, includes: a Cardan joint at the pivot to reduce asymmetrical inertia; modular mounting option; an adjustable end stop; the option to lock all axes; a circular aluminium platform to attach the RCS including the body tube; the possibility to mount a rotary encode on the pitch axis to evaluate the RCS performance; a communications module to initiate a test run and receive data from the rotary encoder.

At the time of drafting this thesis, the complete test stand infrastructure is not finalized. The structure of the test stand with the body tube mounted on top is shown in Fig. 3-6. It is important to distinguish between the rocket coordinate frames and the test stand coordinate frames. The body-fixed coordinate frame of the rocket will originate at the centre of gravity, which will which will be the pivot point as well. The inertial frame on the actuator will originate at the pivot point as well, however this will not be the centre of mass of the test stand. The actuator coordinate frames origin is placed along the symmetry axes at the height of the actuators. For the test stand, this will roughly be where IMU will be placed.

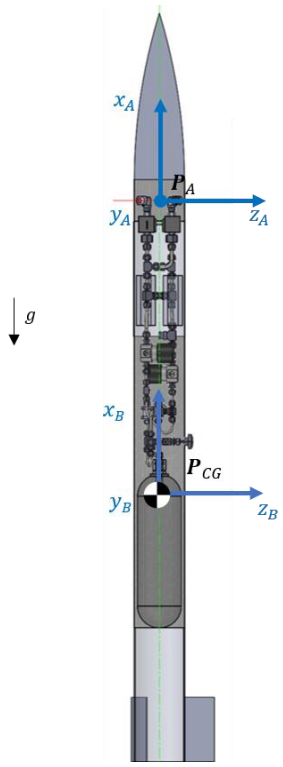


Fig. 3-5: WARR EX-ONE with the body-fixed coordinate frame, B, and actuator frame coordinate frame, A.

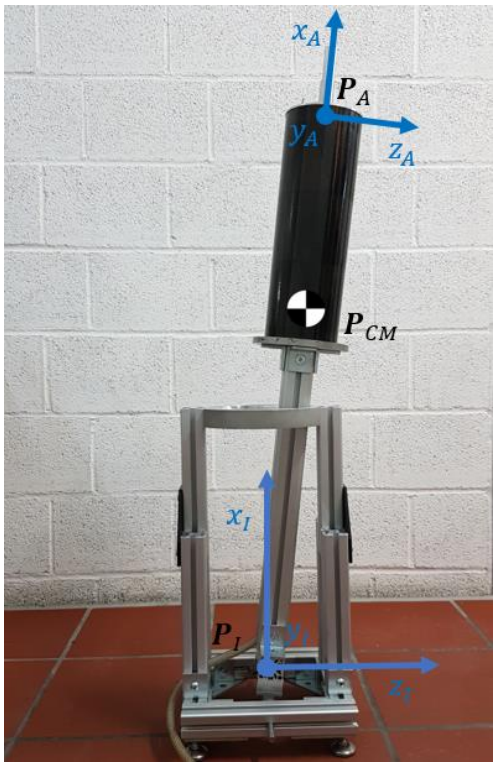


Fig. 3-6: Test stand with the inertial coordinate frame, I, and the actuator coordinate frame , A. Actuators not mounted.

The mass of the pendulum including the RCS was calculated to be $m_0 = 5\ \mathrm{kg}$ [26]. The dynamics of the pendulum will be considered for the model in section 4.1.

The circular structure in Fig. 3-6 will function as an end stop. It will be adjusted to $\theta = 10\ °$, which gives a linear behaviour between gravitational force and deflection using the small angle approximation. Since, the end stop resembles no physical behaviour during a flight test, hitting it will disqualify the test run.

3.1.3 Requirements

A list of 67 requirements [27] were formulated to cover safety-related (SXX) and functional demands (FXX) for the different project phases. A selection of requirements relevant to this thesis are listed below:

Table 3-3: Extract out of the requirements document [27] translated into English.

ID	Description
F01	The RCS shall stabilize the pendulum in the vertical direction around a fixed attitude
F07	The angular deviation from the fixed attitude shall not be more than $1\,°$. In the case of phase 0,1,2,3, this applies after the oscillation of the stabilization process from the starting attitude has subsided
F08	The RCS shall be able to stabilize the pendulum for $t = 15\,s$
F13	The RCS shall control & stabilize the pitch axis of the pendulum
F21	The RCS shall remain operational within a temperature range of $-30\,°C \leq T_E \leq +50\,°C$
F23	The mass of the RCS shall not exceed $m = 4\,\mathrm{kg}$
F24	The RCS shall remain functional after prolonged periods of inactivity of up to one hour
F32	Under no circumstances should the pendulum reach the end stop after the stabilization process has begun
F33	When designing the controller, a phase margin of $\phi_R = 60\,°$ shall be provided in accordance with MIL-F-9490D (USAF) [28]
F34	The overshoot shall not be greater than $\theta_{OS} = 60\,\%$ of the initial error to the commanded trajectory
F35	The root-mean-square-error (RMSE) of the true trajectory to the commanded trajectory shall not exceed $RMSE = 2.5\,°$

Therefore, F35 represents the minimum performance, in terms of $RMSE$, a simulation run shall possess to be deemed successful. Introducing the $RMSE$ as the main performance measure.

3.2 Approach

The unusual circumstances of the project demand for a highly independent, cost-effective and parallelized development and simulation procedure. This is introduced in the next sections.

3.2.1 Development Procedure

The development procedure of the subsystems in the project is parallelized. This enables a subsystem to incorporate design changes suggested by the results of other team members' design progress. This approach works for a frequently communicating and interconnected team. However, it requires the team members to estimate design parameters for the early development stage. This led to the reference values, introduced in Chapter 6, which were used for a preliminary and a main test series. The results of the preliminary and main test series of this work will help team members evaluate importance and ranges of hardware design driving parameters.

3.2.2 Simulation Approach

Since the RCS hardware is still being developed, the development and testing of the controller and filter is performed in a simulation environment. Leveraging the expertise of the WARR and collaborating institutes, a simulation-only approach was chosen as a first step to save time, reduce costs, and provide a baseline for the later software implementation. The chosen simulation environment is Simulink, a graphical simulation environment for modelling, simulating and analysing dynamical systems developed by The MathWorks Incorporated©. The utilized version is `Matlab R2017b SimulinkVersion 9.0` with the `Continuous Toolbox Control, Design Toolbox` and the `Control System Toolbox` installed.

For this approach, first, the test stand dynamics, the actuator model and the controller were implemented in the simulation environment. The required parameters for the models were mostly acquired from literature, datasheets and calculations by the hardware design team. This model was used to tune the controller (see 4.3.1). Then the model was extended by the sensor error model, which introduced the necessity for an appropriate filter algorithm. With the finished model, extensive simulation test runs can be executed, improving the control algorithms and providing valuable information for further design iterations of hardware and software. In [29], a similar approach was performed for the development of an RCS on a suborbital missile.

One drawback of this concept is that the test series will not consider hardware related aspects, which are not covered by the simple mathematical models described in Chapter 4. A software-in-the-loop (SIL) approach might be appropriate and will be considered as the next step to this thesis. This topic will be thematized in Chapter 7.

3.3 Delimitation

As explained in 3.2.1, the workload was divided among the project members to cover the design of the whole control system:

Till Assmann will develop the design of the solenoid valve actuators, the gas tank and the pressure regulation and distribution system for applications on the test stand as well as on the rocket concept [5, 26].

Matthias Beck will develop the three-dimensional model of the inverse pendulum (P2), and subsequently the rocket, to design an appropriate control algorithm [30, 31].

Tim Klose will implement the algorithms developed in this thesis on the Teensy 3.6 microcontroller [32].

Therefore, all topics regarding microcontroller implementation, model fidelity, controller design, optimization and stability analysis, hardware design, trajectory optimization are covered by fellow project members and will not be addressed within this work.

Nevertheless, the whole control loop for 1-DoF, corresponding to P1, will be modelled in Simulink to evaluate the performance of the filter and report back concerns and considerations to team members. The utilized equations to model the respective hardware and software elements will be detailed in the next chapter.

4 Mathematical Simulation Model

To give an accurate evaluation of the filter without a functioning hardware setup, a model of the controller, the actuators, the plant and the sensors as well as a desired reference trajectory is needed.

For modelling in Simulink, the naming conventions defined by the MAAB guidelines [33] are adopted. In order to intuitively differentiate between the different blocks and their functions, the colour palette (see Fig. 4-1) proposed by New Eagle LLCs Simulink Style Guide [34] is used.

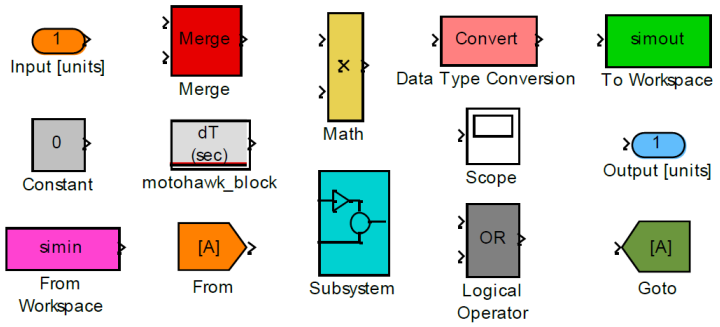


Fig. 4-1: Color palette designed by New Eagle LLC [34]. The `motohawk_block` can be ignored.

The main level of the RCS-model can be accessed by selecting the `RCS1-DoF` block (see Fig. D-8) on the first level and is depicted in Fig. 4-2. The input variables of the model can be initialized either by the MATLAB script `initialize_RCS.m` or through the mask of the respective subsystem. Further information on the initialization and handling of the model can be found in the appendix B.

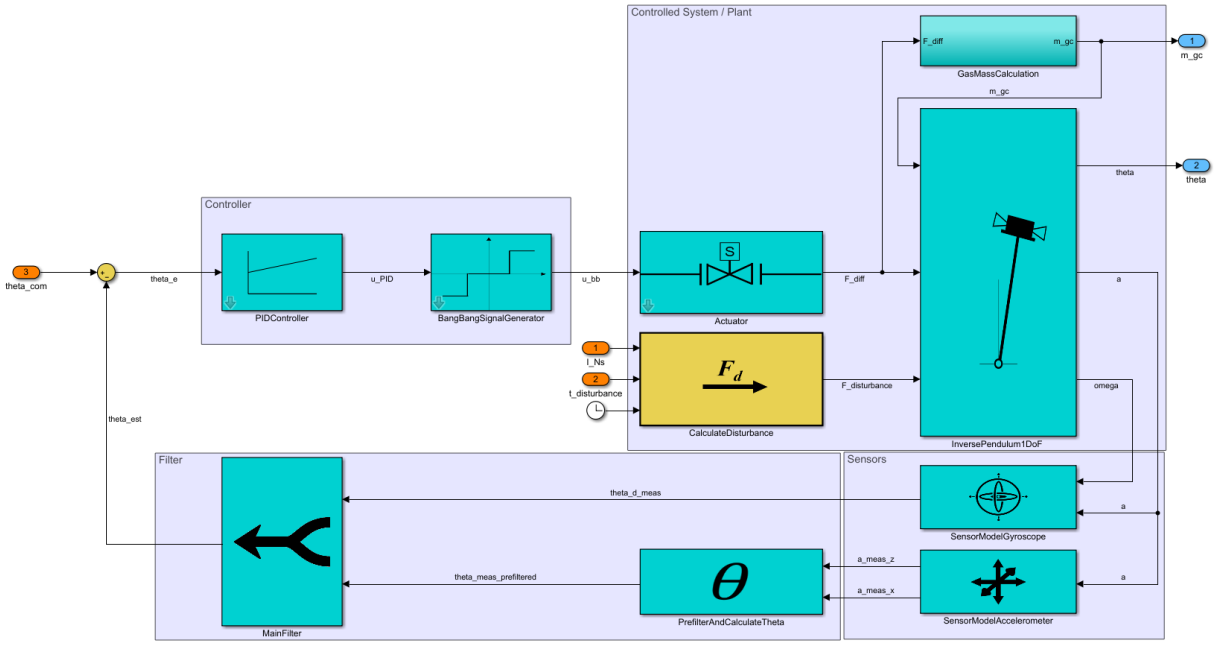


Fig. 4-2: The whole control loop `RCS1DoF`, including controller, actuator, pendulum, sensor models and the filter, resides at the second level of the Simulink model. The trajectory is generated at the first level.

Every block in Fig. 4-2 is referred to as a subsystem of the control loop. To avoid confusion, the model presented in Fig. 4-2 will be referred to as the main model, while a reduced model and an example model will be introduced in section 4.3.1 and 5.1.2, respectively.

In the following sections every subsystem except the filter is derived.

4.1 Inverse Pendulum Mechanics

The test stand is modelled as a simple 1-DoF inverse pendulum without aerodynamical damping or friction losses. This simplification stems on the assumption, that for the test stand, aerodynamical damping forces and friction losses are negligible compared to the gravitational and actuator force. The gravitational acceleration acts in the negative direction of the established x-axis of the inertial frame and acts on the centre of mass at the location P_{CM}, effectively reducing the rigid body of the entire system to a mass point with the mass m. This includes the structural weight of the inverse pendulum and the RCS. A mass calculation was performed in [5], resulting in a total weight of $5\,\mathrm{kg}$, including a $10\,\%$ margin. The total actuator force, F_{diff}, is defined as the difference between the forces acting on the pendulum produced by the two solenoid valve actuators, detailed in section 4.4. Just as the gravitational force, the actuators produce a moment around the y-axis of the inertial frame, y_I, with the lever arm l_A. This circumstance is depicted in Fig. 4-3.

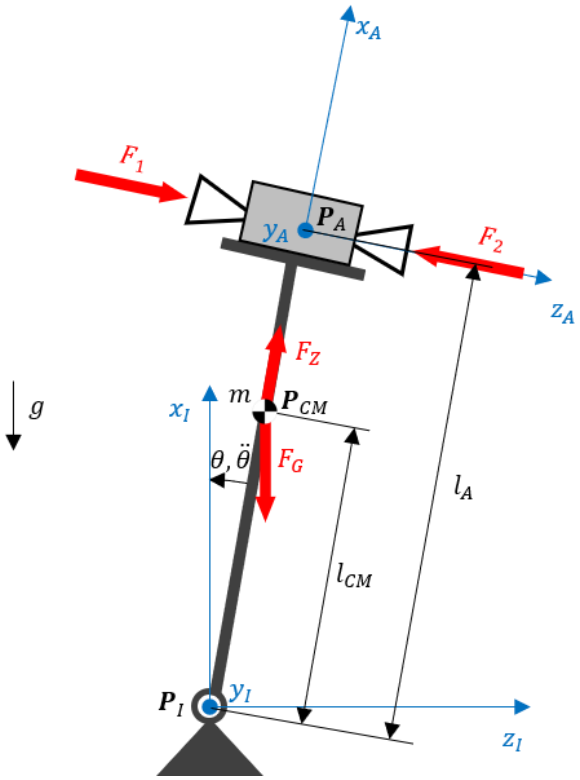


Fig. 4-3: Free cut of the test stand modelled as a 1-DoF inverse pendulum.

By either applying a moment equilibrium around the origin of the inertial frame, $\boldsymbol{P_I}$, or the Lagrange 2 formalism, the differential equation (DE) for the pendulum can be obtained. The DE can be formulated as:

$$\ddot{\theta} = \frac{l_A}{l_{CM}^2 m} F_{diff} - \frac{(-g)}{l_{CM}} \sin(\theta) \qquad \textbf{Eq. (4–1)}$$

with

$$F_{diff} = F_1 + F_2 \; and \; m = m_0 = const. \qquad \textbf{Eq. (4–2)}$$

The point mass of the system will be assumed to be constant in the main test series, since the change in mass due to gas loss will be covered by the mass variation range explained in section 6.2.3.2. However, a switch was implemented to activate mass reduction due to gas loss (see appendix Fig. D-9). Eq. (4–19) was used to calculate the gas consumption for a simulation run.

As established in section 3.1.2, the end stop of the test stand was set to $10\,°$. Therefore, the range of motion of the inverse pendulum can be formulated as:

$$-\theta_{ES} \leq \theta \leq \theta_{ES}, \qquad \textbf{Eq. (4–3)}$$

and the angle will always be inside the range of the small angle approximation. This allows to omit the sine. Therefore, the linearized equation of motion can be Laplace-transformed into the frequency domain formulating the transfer function Eq. (4–4) with the Laplace frequency domain variable s.

$$G_{\Theta F_{diff}}(s) = \frac{\Theta(s)}{F_{diff}(s)} = \frac{\frac{l_A}{l_{CM} m}}{s^2 - \frac{g}{l_{CM}}}. \qquad \textbf{Eq. (4–4)}$$

However, in the model, the DE, Eq. (4–1), is used to obtain the angular acceleration and therefore translational acceleration on the z-axis of the actuator frame. The A-frame originates in the middle of the structural plate approximately on the same plane as the force application point of the thrusters and shall be identical to the coordinate frame of the IMU. The measured translational acceleration, $a_{meas,A,z}$, will adopt the binary characteristics caused by the Two-Step controller, which will be introduced in section 4.3. The resulting issue for the state estimation and how to handle it is addressed in Chapter 5.

The centrifugal force, F_Z, is computed with:

$$F_{Z,x} = m l_{CM} \dot{\theta}^2. \qquad \textbf{Eq. (4–5)}$$

Dividing F_Z by the mass and adding the gravitational acceleration component gives the acceleration measured by the sensor along the x-axis of the A-frame, which is referred to as $a_{meas,A,x}$. As this is the 1-DoF model and the test stand is fixed in its place, the gyroscopic force and the Coriolis force influence on the accelerometer measurement is neglected.

For the implementation in Simulink it is important to use the `Integrator, Second-Order` block, which can be found in the `Continuous Toolbox` [35]. With this block the condition can be set to set the first integration, which is the angular speed, to zero if the end stop is reached. However, reaching the end stop during the stabilization process for the real test stand will produce a reaction force which is not included in the simulation. Therefore, hitting the end stop will disqualify any simulation run. Further, this reaction force would not be present on the rocket. This deliberation led to the formulation of requirements F32 and F34 in Table 3-3.

In summary, the `InversePendulum1-DoF` will produce the attitude of the pendulum, $\theta(t)$, the accelerations along the x- and z-axis acting on the sensor and the angular velocity around the y-axis of the sensor frame as outputs from the actuator input force, F_{diff}. The attitude, $\theta(t)$, may also be referred to as the true trajectory, not to be confused with the commanded trajectory, θ_{com}, detailed in the next section.

4.2 Trajectory Generation

The course of the commanded trajectory, θ_{com}, also referred to as desired trajectory, depends on the selected trajectory mode, T_{mode}. The operator of the simulation is presented with three different trajectory modes. All trajectories possess a simulation time of $t_{sim} = 16\,s$. This originates from requirement F08 and an additional second of filter buffer time, t_b. The buffer time was chosen to be greater than the rise time, t_r, of the filters which depends on the filter input parameters. This interrelation is formulated in section 5.1 Eq. (5–5) and is discussed in 5.4.2. Further, all trajectories can be described as one or multiple step inputs at specific points in time. The three commandable trajectories are formulated in the following section:

$$\theta_{com}(T_{mode} = 1, t): \begin{cases} \theta_{ES}, & for\ t < t_b = 1\ [\mathrm{s}] \\ 0, & for\ t \geq t_b = 1\ [\mathrm{s}] \end{cases}, \qquad \textbf{Eq. (4–6)}$$

with

$$F_{disturbance}(T_{mode} = 1, t): \begin{cases} 0, & for\ t < 8.0\ [\mathrm{s}] \\ 20, & for\ 8 \leq t \leq 8.1\ [\mathrm{s}]. \\ 0, & for\ t > 8.0\ [\mathrm{s}] \end{cases} \qquad \textbf{Eq. (4–7)}$$

Trajectory mode one possesses the particularity that a perturbation force, $F_{disturbance}$, will try to destabilize the pendulum for a defined period. The course of the commanded trajectory and the perturbation is visualized in Fig. 4-4 and Fig. 4-7. $F_{disturbance}$ will remain zero over the course of the other two trajectories.

Trajectory mode two will include a step input after eight seconds:

$$\theta_{com}(T_{mode} = 2, t): \begin{cases} \theta_{ES}, & for\ t < t_b = 1\ [\mathrm{s}] \\ 0, & for\ 1 \leq t < 8\ [\mathrm{s}]. \\ 5, & for\ t \geq 8\ [\mathrm{s}] \end{cases} \qquad \textbf{Eq. (4–8)}$$

This shall simulate a manoeuvre, which can be expanded to a gravity turn or a side step manoeuvre. The course of the commanded trajectory is visualized in Fig. 4-5.

Trajectory mode three is the most significant commanded trajectory to this work since it is the least complex and is used to evaluate the performance of the filter algorithms.

$$\theta_{com}(T_{mode} = 3, t): \begin{cases} \theta_{ES}, & for\ t < t_b = 1\ [\mathrm{s}] \\ 0, & for\ t \geq t_b = 1\ [\mathrm{s}] \end{cases}, \qquad \textbf{Eq. (4–9)}$$

The course of the commanded trajectory is the same as for trajectory mode one, but without any perturbation and is visualized in Fig. 4-6.

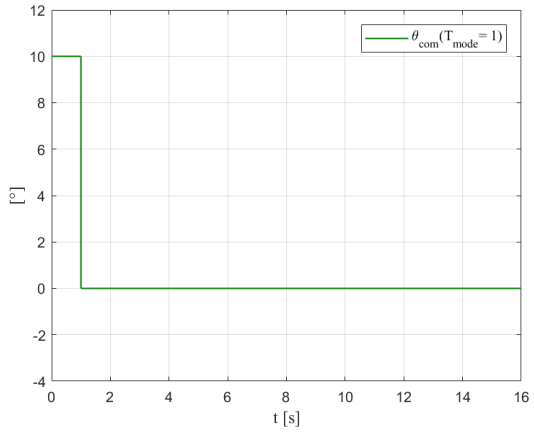


Fig. 4-4: Commanded trajectory for mode one.

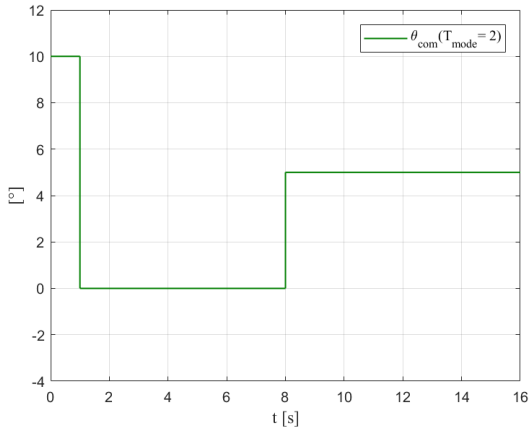


Fig. 4-5: Commanded trajectory for mode two.

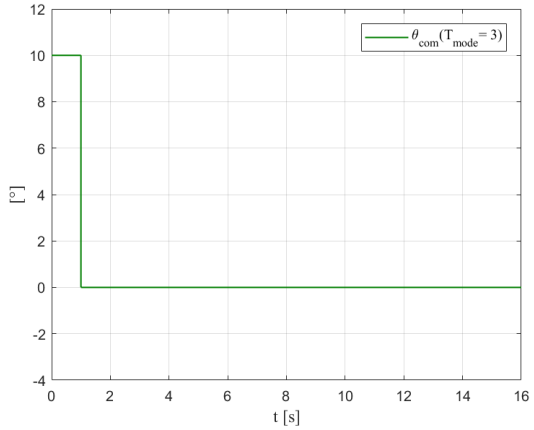


Fig. 4-6: Commanded trajectory for mode three.

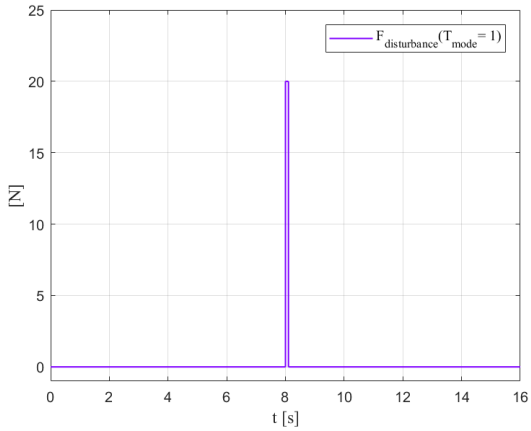


Fig. 4-7: Perturbation in mode one.

To increase the time efficiency of the simulation run it is recommended to disable the `zero-crossing detection` in the `Integrator, Second-order` block. Excessive simulation times for second order systems with constant position or attitude, as it is the case for the buffer time, is a well-known problem documented in [36].

To enable the pendulum to achieve the commanded trajectory, a controller needs to be designed which is subject of the next section.

4.3 Controller

The development of an optimal control algorithm is not the objective of this thesis. Therefore, the following control algorithm represents just one possible solution. The focus for this preliminary controller design is simplicity. In [29] a proportional controller in combination with duty cycle limiter and pulse width modulation (PWM) is suggested for stabilization around the roll axis. However, the inverse pendulum undamped PT2 dynamics can arguably be classified as a more unstable system than the rolling cylinder in [29] which is modelled as a PT1 with a certain damping coefficient. Therefore, a faster controller is necessary while also fulfilling steady state accuracy implied by the requirements F07 and F35 in Table 3-3.

Furthermore, the function for the duty cycle limiter block which transforms the required thrust force into a solenoid valve duty cycle required for the PWM, is acquired by using linear regression determined through PWM single-stream testing of the propulsion system [29]. The necessary facilities for such kind of testing is not available to the WARR and was therefore not an option.

This thought process led to controller design, consisting of two parts, which will be discussed in the next sections.

4.3.1 PID Controller

The proportional-integral-derivative controller (PID controller) is a wide-spread control loop feedback mechanism [37]. The PID controller applies a correction to the continuously calculated error, between the true trajectory and the commanded trajectory, based on the proportional, integral and derivative terms.

For this work, the error in the time-continuous domain is calculated with

$$\theta_e(t) = \theta_{com}(t) - \theta(t). \qquad \textbf{Eq. (4–10)}$$

Therefore, the complete time-continuous PID control function can be expressed as

$$u_{PID}(t) = K_p\theta_e(t) + K_i \int_0^t \theta_e(t')dt' + K_d \frac{d\theta_e(t)}{dt}, \qquad \textbf{Eq. (4–11)}$$

where the K_p, K_i, K_d, represent the strictly non-negative coefficients for the proportional, integral and derivative terms respectively [37]. Therefore, the Laplace transform to the frequency domain gives the transfer function

$$G_{U_{PID}\Theta_e}(s) = \frac{U_{PID}(s)}{\Theta_e(s)} = K_p + K_i\frac{1}{s} + K_d s. \qquad \textbf{Eq. (4–12)}$$

The following paragraph will try to give an intuitive explanation for the underlying mathematics of the controller: The proportional term evaluates the current state of the system and addresses it with a weight K_p; the integral term evaluates the past state of the system and addresses it with a weight K_i which guarantees steady state stability; and the derivative term evaluates the future state of the system and addresses it with a weight K_d which allows the system to respond faster to abrupt changes in trajectory. For the 1-DoF case, the PID control algorithm will only need the attitude as a state feedback, which will be estimated by the filter developed in Chapter 5. The angular velocity will not be fed back.

For the tuning of the PID controller gains, the Simulink `PID Tuning Tool` [38] from the `Control Design Toolbox` was used on a reduced ideal model of the system. The reduced model includes the PID controller, the actuator model and the inverse pendulum model. The `PID Tuning Tool` calculates the optimal PID-gains, with the `pidtune(system,'PID')` algorithm [39], patented by The MathWorks Inc.®, for the linearized closed-loop plant. Further, the tool allows to adjust the gains to obtain the desired phase margin or aggressiveness of the control algorithm, respectively. For this project, a high phase margin of 60 °, according to requirement F33, is targeted for the reduced model which may be decreased by artificially added time delays causing a phase shift in the main model or unanticipated time delays in the real test stand.

After the PID control algorithm has calculated a u_{PID} value, the signal is delayed by $t_{PIDdelay} = 0.01$ s with a `Transport delay` block to simulate the latency caused by

the microcontroller calculation time. The value for the delay time was measured by the implementation and software team [32]. Since the delay time may increase when scaling up to multiple degrees of freedom, the examination of this parameter will be a primary focus of the evaluation in section 6.3.

The output of the PID controller cannot be processed by the actuator interface, which only accepts a binary input. Therefore, a Two-Step controller as an alternative to a PWM controller is introduced in the next section.

4.3.2 Two-Step Controller

The Two-Step controller (German: Zweipunktregler), more commonly referred to as bang-bang controller (BB controller), is a, for some applications time-optimal, feedback controller that abruptly switches between two states [40].

As previously mentioned, the BB controller was chosen as the second part of the control algorithm due to its simplicity and convenience for this application. No testing needs to be performed to find an appropriate duty cycle for the respective u_{PID} value, as for the PWM controller. In theory, the output of the BB controller, u_{bb}, can obtain one of two states depending on the input condition [40]. Transferred to this work, the BB controller output shall control both actuators depicted in Fig. 4-3, with the condition that only one actuator can be active at the same time for the 1-DoF case. Additionally, a range of inputs u_{PID} shall be defined in which no actuator is active. This threshold will be labelled u_{bb0} and was implemented to save gas. This assumption will be verified in 6.2.2. The resulting conditions for u_{bb} can be formulated as

$$u_{bb}: \begin{cases} 1, & for\ u_{PID} > u_{bb0} \\ 0, & for\ -u_{bb0} \leq u_{PID} \leq u_{bb0} \\ -1, & for\ u_{PID} < -u_{bb0} \end{cases}. \qquad \textbf{Eq. (4–13)}$$

The algorithm is implemented in Simulink with the `MATLAB function` block. By selecting `discrete` as the `update method` and setting the `sample time` to chosen actuator frequency, f_{bb}, the output, visualized in Fig. 4-8, becomes unsteady.

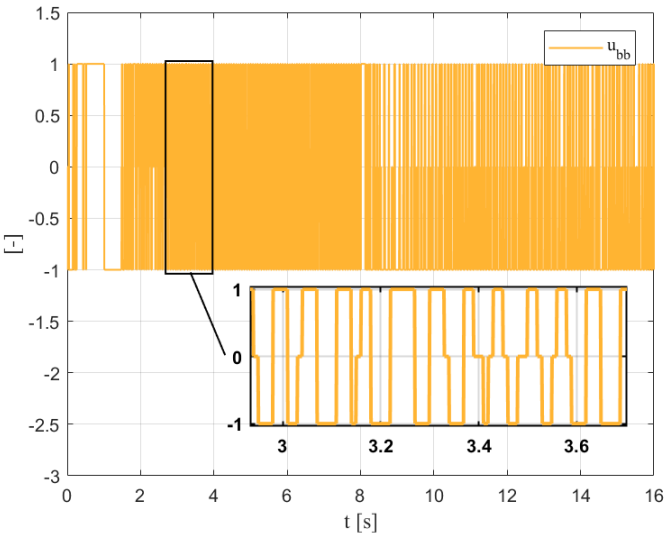


Fig. 4-8: Bang-bang controller time-discrete output signal for $f_{bb} = 100$ Hz and $u_{bb0} = 0.1$.

Considering Eq. (4–13), the bang-bang controller output is comparable with the Heaviside step function [41], which is classified as a stiff problem [42]. Therefore, using an explicit solver, for example the Simulink default setting `ode45`, is time inefficient. Hence using the implicit solver, `ode15s`, is recommended for this simulation [43]. Further, the Simulink stability analysis tool is not applicable for the BB controller. Analysing the stability of the closed loop for the final controller design will be a task for future work on this project by the control team [31].

In this section, the PID controller output was successfully transformed into the $u_{bb}\,\epsilon\,[-1;0;1]$ signal accepted by the two actuators. In this model $u_{bb}=-1$ will activate the actuator facing the opposite direction of the actuator activated with an $u_{bb}=1$ (see Fig. 4-3). However, the actuator model detailed in the next section will give the output of both actuators combined, labelled F_{diff}.

4.4 Actuator Modelling

In 3.1.1 the cold gas thruster RCS, designed in [5], was presented. To model the utilized actuators the valve and the nozzle will be considered in this section.

The valve for the first design iteration will be the MHE4 solenoid valve from Festo AG & Co. KG. The product datasheet reveals an ON-switching / OFF-switching time of $t_{st}=0.0035\ s$ and a maximum switching frequency of $f_{max}=210\ \mathrm{Hz}$ [23]. Further, the hardware design team calculated a maximum thrust of $F_{max}=7\ \mathrm{N}$ with the current nozzle design [26]. This provides enough information to model the cold gas solenoid valve actuators as a PT1 element. The simplification of a solenoid valve transient behavior to a PT1 element is suggested by [29]. A similar but more sophisticated approach can be found in [44]. The first order differential equation for the described actuator can be formulated as

$$\tau\frac{\partial F_{diff}(t)}{\partial t}+F_{diff}(t)=Ku_{bb}(t), \qquad \textbf{Eq. (4–14)}$$

with the corresponding transfer function in the frequency domain

$$G_{F_{diff}U_{bb}}(s)=\frac{F_{diff}(s)}{U_{bb}(s)}=\frac{K}{\tau s+1}, \qquad \textbf{Eq. (4–15)}$$

Including the time constant

$$\tau=\frac{t_{st}}{-\ln\left(1-\frac{F(t=t_{st})}{F_{max}}\right)}=7.6*10^{-4}\ [\mathrm{s}], \qquad \textbf{Eq. (4–16)}$$

and the gain

$$K=F_{max}=7\ [\mathrm{N}]. \qquad \textbf{Eq. (4–17)}$$

Furthermore, the thrust reached after the switching time, t_{st}, could not be extracted from the datasheet [23] and was therefore, according to [44] assumed to be:

$$F_{st}=\frac{F(t=t_{st})}{F_{max}}*100=99\ [\%]. \qquad \textbf{Eq. (4–18)}$$

The transient behavior of the modelled actuator for a step input is shown in Fig. 4-9.

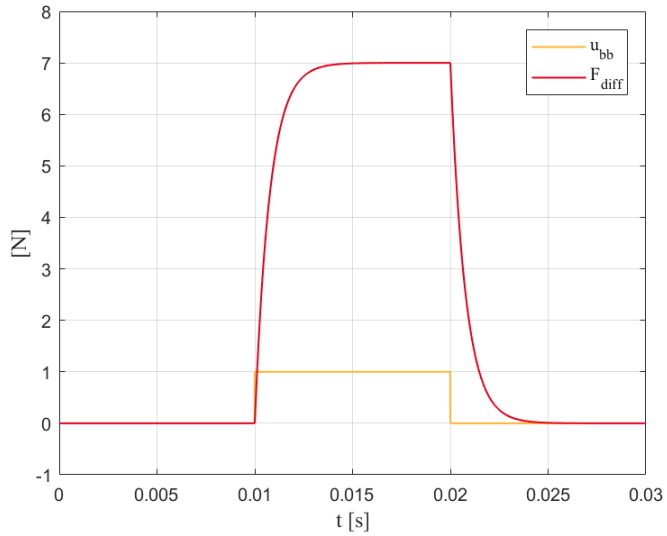


Fig. 4-9: Transient behavior of the modelled solenoid valve MHE4 for a step input u_{bb}.

If the utilized valve changed, the `Actuator` block in the Simulink model can be adjusted if the values for the parameters described in this section are known. Additionally, there is the option to add a deadtime to the actuator. This was realized through a `Transport Delay` block and can be adjusted by the variable $t_{actdelay}$.

The cold gas consumption of the RCS during a test run is calculated with average actuator exhaust velocity v_{act}, calculated in [5]. Therefore, the cold gas consumption is formulated as

$$m_{gc}(t) = \frac{1}{v_{act}} \int_0^t \left| F_{diff} \right| dt, \qquad \textbf{Eq. (4–19)}$$

within the `GasMassCalculation` block. The total gas consumption will depend on the filter selection and u_{bb0}. Results will be analyzed in 6.2.2.

The total actuator thrust, F_{diff}, will act on the inverse pendulum model, described in 4.1, resulting in the true trajectory, $\theta(t)$, the translational acceleration and the angular velocity measured by the MEMS sensors. These will add deterministic and stochastic errors to the signal which will be detailed in the next section.

4.5 Sensor Model

The IMU used for the RCS is the InvenSense MPU-6050 [14]. It contains a MEMS vibrating-beam accelerometer as well as a MEMS vibratory gyroscope in a single chip. The sensor will use the I2C-bus to interface with the Teensy 3.6. To model the accelerometer and the gyroscope for the 1-DoF case, stochastic, also referred to as probabilistic or statistical, and deterministic errors are considered. All the necessary parameters to model the random error processes can be obtained from an Allan Variance analysis. However, the necessary equipment to perform it, such as an optical table with minimum deflection, are not available to the WARR.

Therefore, the model will only contain the following types of random error processes [45] which could be gathered from [21, 46]:

Quantization Noise: This noise term is introduced into an analog signal after it is converted into a digital signal. The measurement is either truncated or rounded because of the set range and limited resolution of the sensor and will result in the quantization error. This error possesses the same unit as the measurement.

White Noise / Angle Random Walk: This noise term is characterized by a white-noise spectrum on the sensor output. It is usually marked with the letter w. In case of the gyroscope, the measured rate is integrated and therefore results in a random drift in the angle, which is dubbed Angle Random Walk. It is commonly unitized with, $(°/s)/\sqrt{\text{Hz}}$ for the gyroscope, or $(\text{m/s}^2)/\sqrt{\text{Hz}}$ for the accelerometer. However, for the sensor error model, this stochastic process will be modelled as white gaussian noise with the standard deviation (STD), σ, measured by [21, 46], and zero mean. The STD possesses the same unit as the measurement.

$$w = \mathcal{N}[0, \sigma] \qquad \textbf{Eq. (4–20)}$$

Bias Instability: This noise type is caused by the electronics or related components that are receptive for random flickering. It evaluates how the bias, or initial offset, of the measurement changes over a specified period at constant temperature. This is typically presented in units of $(°/s)/\text{hr}$ for a gyroscope. Essentially, this drift in bias cannot be accounted for through calibration as it is the case with the Static Bias. In the scenario of this project, the drift in Bias over one hour (see F24), is $-0.035\,°/s$ for the gyroscope, according to data from [46]. This error term will be referred to as the residual bias, which is marked with the index RB in the subscript. It will lead to a considerable drift in the angular position of $-0.56\,°$ over the course of the simulation time, if state estimation is performed by solely integrating the gyroscope measurement, visualised in Fig. 4-11. This proposes the first major concerns of this thesis and will be addressed by the filters developed in Chapter 5. The accelerometer value is not integrated and therefore the effect of Bias Instability is neglected.

The deterministic errors were gathered from the product datasheet [14] and measurements by [32]:

Static Bias: For a given physical input, the sensor outputs a measurement which is offset by the bias [47]. By measuring the output of the sensors at a fixed known position and subsequently averaging the data, the Static Bias can be obtained. Therefore, it can be compensated. This process is referred to as calibration. For this model the bias is denoted with the index B in the subscript and possesses the same unit as the measurement.

Scaling: For this work, scaling shall include a multitude of multiplicative, linear errors. Quadratic scale factor terms and even higher order terms (hot) are excluded. For 3-DoF with the measured states $\boldsymbol{x}_{meas,M}$, exclusively considering multiplicative errors, the following expression is formulated:

$$\boldsymbol{x}_{meas,M} = \begin{bmatrix} M_{SF,xx}M_{TS,x} & M_{MA,xy} & M_{MA,xz} \\ M_{MA,yx} & M_{SF,yy}M_{TS,y} & M_{MA,yz} \\ M_{MA,zx} & M_{MA,zy} & M_{SF,zz}M_{TS,z} \end{bmatrix} \boldsymbol{x}_{true} + \cancel{hot} = \boldsymbol{M}\boldsymbol{x}_{true} \qquad \textbf{Eq. (4–21)}$$

With the scale factor, M_{SF}, on the diagonal of matrix $\boldsymbol{M}$ and the misalignment representing the off-diagonal of matrix $\boldsymbol{M}$. The scale factor error is defined as the relation between input and output. The scale factor distorts the measurement in a way where the output is proportional to the input but scaled [47]. It will be unitized with

%. The misalignment error is caused by fabrication or installation flaws. For 1-DoF it can be simplified to another multiplicative error term unitized with %, identical to the scale factor. Further, it was decided to incorporate an additional multiplicative term related to a change in temperature, due to high temperature-gradient a missile application might be facing (see F21). It will be labelled temperature sensitivity and possesses the unit $\%/°\mathrm{C}$. In Eq. (4–21) it is marked with the index TS in the subscript.

Higher order scale factor terms (hot) as well as nonlinear scaling effects and cross coupling will be neglected for this work. For information on high fidelity sensor models refer to [48].

Acceleration Sensitivity: Especially consumer grade IMUs are subject to a change in the bias depending on how the sensor experiences acceleration. This is most common in MEMS-gyroscopes, since their sensing elements include moving parts, an acceleration or vibration will be mistakenly detected as the angular speed (see 2.2.2). This effect can be modelled and removed from the measurements and is often included in the IMU signal condition stage before output of the measurements [47]. For this scenario, the Acceleration Sensitivity is unitized with $°/s/g$ and will be included in the gyroscope model with the index AS in the subscript.

Other probabilistic, such as the Rate Random Walk, or deterministic errors, such as the Rate Drift Ramp or the turn-off / turn-on bias error, were excluded from the sensor error model since their impact is negligible over a simulation time of $16\,s$. The theoretical background for the Allan variance analysis will not be explored in the frame of this thesis, since it is not needed for the simplified sensor error models. For more information on this topic refer to [45].

4.5.1 Gyroscope Error Model

The underlying dynamic behaviour of a gyroscope, described in 0, can be modelled as a mass-spring-damper system or PT2 element [49]. However, the datasheet [14] revealed no information about the relative damping required for modelling such a system. Therefore, the gyroscope will be modelled as a PT1 element, also referred to as a Low Pass filter, with the natural frequency $\omega_{n,G}$. The natural frequency is equivalent to the drive frequency from [14] as explained in 2.2.2. The corresponding continuous-time transfer function in the frequency domain for the first order Low Pass filter is given in Eq. (4–22).

$$G(s) = \frac{\omega_{n,G}}{s+\omega_{n,G}} \quad with\ \omega_{n,G} = 30000\ [\mathrm{Hz}] \qquad \textbf{Eq. (4–22)}$$

However, with such a high natural frequency the time delay caused by the gyroscope dynamics is arguably negligible.

Now, the different error terms are added to the model. Fig. 4-10 illustrates the different error terms used for the gyroscope model and in which order they are introduced.

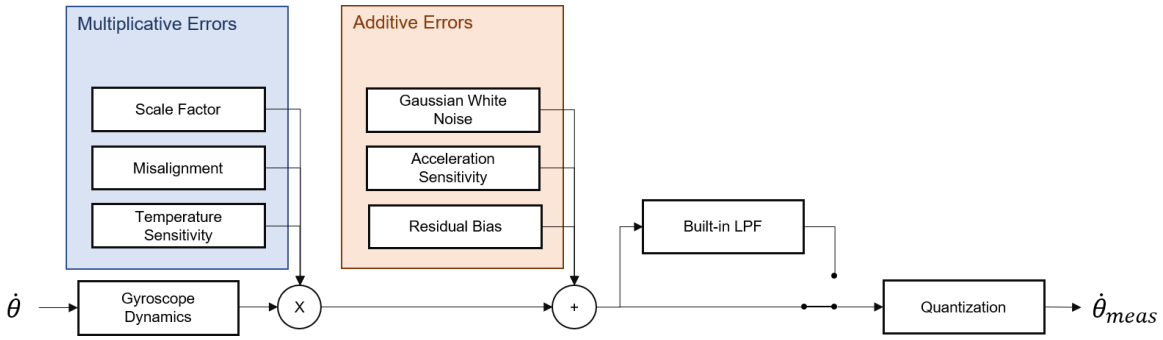


Fig. 4-10: Gyroscope scheme.

The multiplicative error terms are implemented as coefficients. For the reference values they will be set to 1, therefore assuming no scaling. For the evaluation, in section 6.3, they will be varied over range specified by the datasheets [14]. The temperature sensitivity is evaluated by changing the environmental temperature, T_E, over the range specified by the functional requirement F21.

For the additive errors, the static bias was not included since it is assumed that it can be compensated by the calibration. However, a residual bias term is added to account for the previously discussed bias instability and errors in the calibration. The resulting drift in angle is shown in Fig. 4-11. The acceleration sensitivity would usually refer to acceleration along the corresponding axis. However, for the 1-DoF case there is no acceleration towards the y-axis and therefore the acceleration towards the z-axis is chosen which should behave similar to the y-axis acceleration in the 3-DoF case. The noise process for the MPU 6050 was determined by [21] and [46] to be white gaussian in nature. Therefore, it can be implemented in Simulink with the `MATLAB function` block including the function `random('norm',0,STD)`. The corresponding standard deviation, $\sigma_{G,y}$, was measured by the implementation team [32]. It is comparable to the values measured by [21]. The raw measurement data is shown in Fig. 4-12.

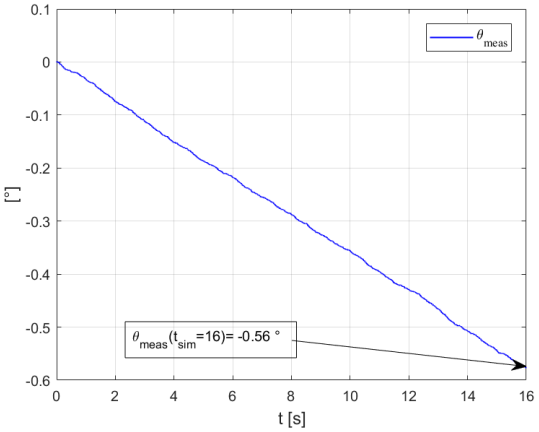


Fig. 4-11: Drift in the y-axis angle due to integration of the residual bias.

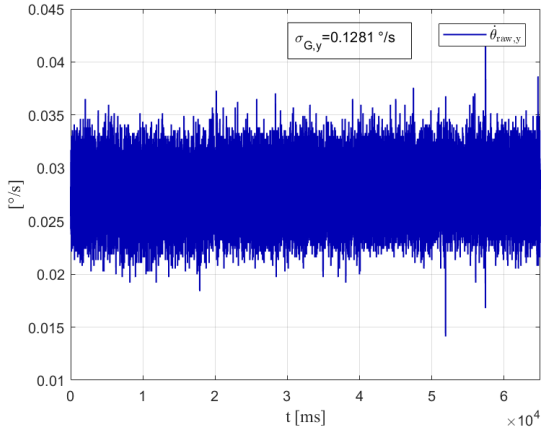


Fig. 4-12: Raw MPU-6050 gyroscope y-axis measurement data of the with a sample rate of $f_{sensor} = 1000$ Hz.

To summarize, the simplified error terms for the gyroscope can be formulated as

$$\dot{\theta}_{meas} = M_{SF,yy} M_{TS,y} M_{MA} \dot{\theta}_{true} + w_{G,y} + \omega_{AS,G,y} a_z + \omega_{RB,G,y}. \quad \textbf{Eq. (4–23)}$$

The static bias term is excluded since it is assumed, that it can be compensated through calibration.

After the additive errors, there the MPU-6050 provides the user with the option to filter the measurements by a Low Pass filter. For the evaluation, this filter will not be used.

At last, the analog signal is converted to a digital signal. This produces the quantization error. It will depend on the measurement range, $\boldsymbol{\omega}_{range}$. If higher angular velocity needs to be measured, less decimal digits will be available introducing an error term resulting from limited resolution. This error may have a noticeable effect on the output if the maximum range is further increased. However, for the reference ranges in 6.1.1 result in a negligible quantization error (see appendix Section E.2).

The complete Simulink model of the gyroscope can be found in the appendix Fig. D-11.

4.5.2 Accelerometer Error Model

The MPU-6050 datasheet contains no information about the natural frequency of the accelerometer. Therefore, it is not included in the accelerometer Error Model. The z-axis accelerometer measurement is modelled similar to the x-axis accelerometer measurement.

Fig. 4-13 illustrates the different error terms used for the accelerometer model and in which order they are introduced.

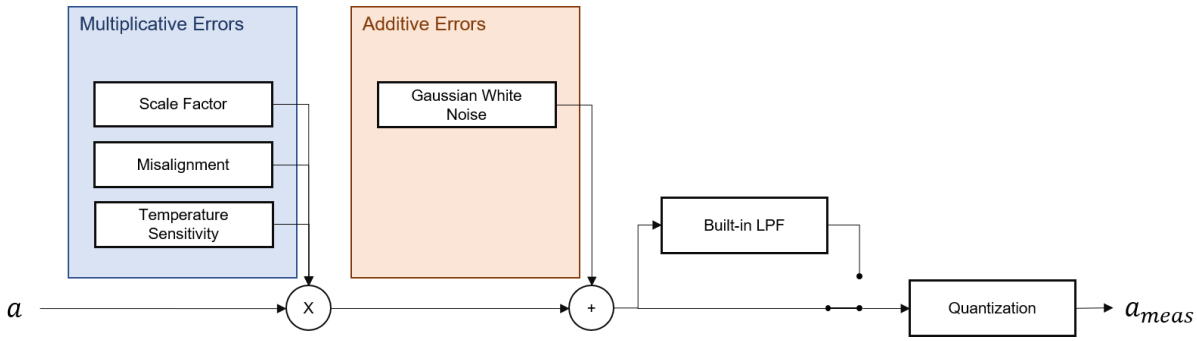


Fig. 4-13: Accelerometer scheme.

For the multiplicative error terms, the same as for the gyroscope holds true. For the reference values, they will be set to 1, therefore, assuming no scaling. For the evaluation in section 6.3, they will be varied over range specified by the datasheets [14].

The accelerometer error model will only include the gaussian white noise as an additive error. Like the gyroscope model, it can be implemented in Simulink with the `MATLAB function` block including the function `random('norm',0,STD)`. The corresponding standard deviations $\sigma_{A,x}$ and $\sigma_{A,z}$ were measured by the implementation team [32]. They are comparable to the values measured by [21]. The raw measurement data is shown in the graphs beneath.

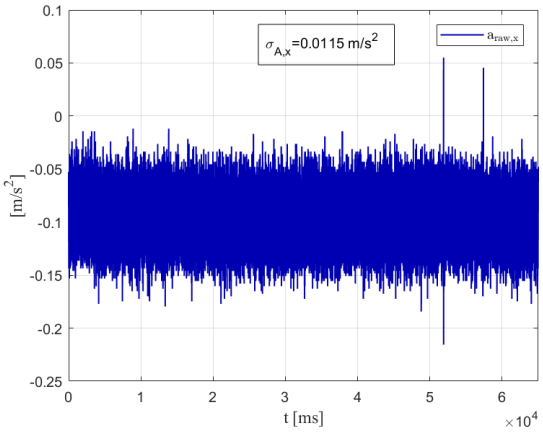


Fig. 4-14: Raw MPU-6050 accelerometer x-axis measurement data of the with a sample rate of $f_{sensor} = 1000\ \mathrm{Hz}$.

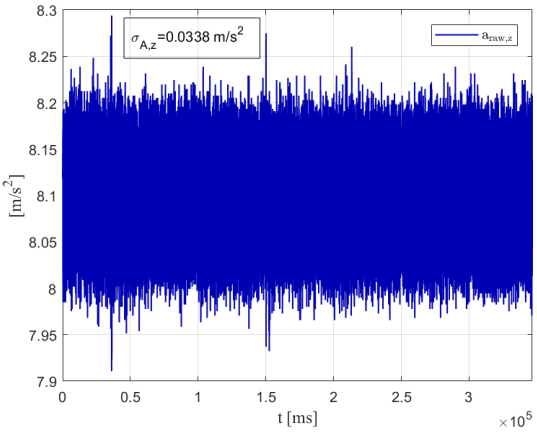


Fig. 4-15: Raw MPU-6050 accelerometer z-axis measurement data with a sample rate of $f_{sensor} = 1000\ \mathrm{Hz}$.

Since a residual bias in the accelerometer measurement would not be integrated, it will not result in a drift in the estimated angle. Therefore, no residual bias is included.

To summarize, the simplified error terms for the gyroscope can be formulated as

$$a_{meas} = M_{SF} M_{TS} M_{MA} a_{true} + w_A, \qquad \textbf{Eq. (4–24)}$$

for each of the measurement axes respectively. The static bias term is excluded since it is assumed, that it can be compensated through calibration.

The in-built LPF and quantization is handled like in the gyroscope model.

The most significant concern about the accelerometer lies in the z-axis measurement, which will acquire an undamped acceleration proportional to the total actuator force F_{diff}. Because of the binary nature of the BB controller, see Fig. 4-8, the acceleration on the z-axis can be described as regularly recurring and alternating, shown in Fig. 4-16. The pure gravitational acceleration on the z- and x-axis would allow for an adequate estimation of the attitude Eq. (5–4). However, the relatively high-g distortion on the z-axis complicates this process.

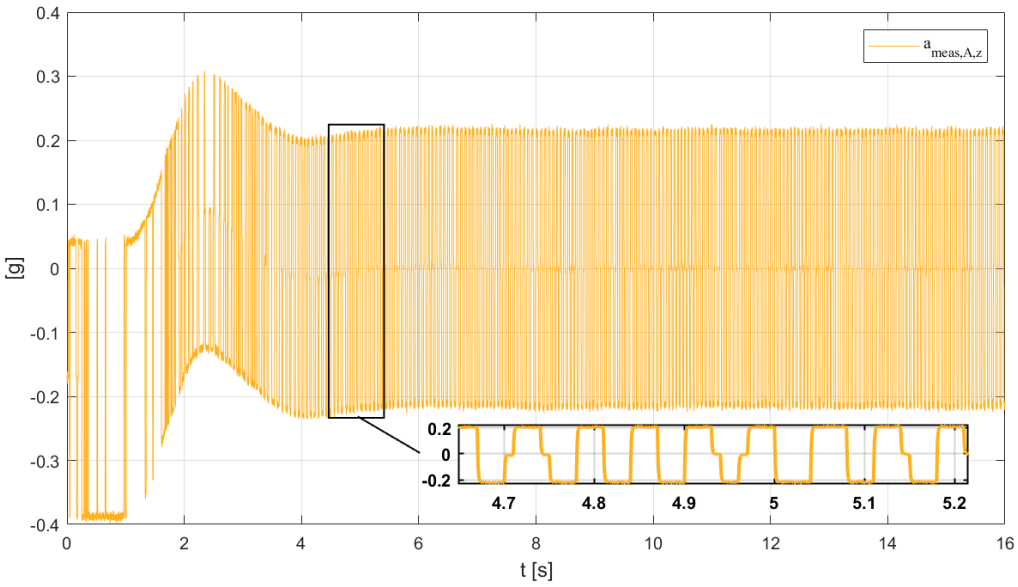


Fig. 4-16: Accelerometer z-axis measurements.

This proposes the second major concern regarding the measurement information. Therefore, the research question RQ1 can be reformulated to: Is there a filter able to estimate the attitude from the distorted data of Fig. 4-11 and Fig. 4-16? This will be the primary focus of the next chapter.

The complete Simulink model of the accelerometer can be found in the appendix Fig. D-12.

5 Filter Development

The measurement data provided by the MEMS-sensors, detailed in section 4.5, is noisy, biased and subjected to relatively high-g dynamics caused by the discrete nature of the two-point-controller. To provide an accurate state estimation for the controller, a suitable filter must be designed. The main tasks of the filter shall be to prevent the integrated gyroscope measurement from drifting and to smooth the uniform high amplitude peaks in the z-axis accelerometer measurements. For this purpose, a prefilter, which will be applied on the accelerometer measurements, is introduced in section 5.1. Afterwards, the theoretical background and implementation of the two main filter concepts, a Complementary filter and discrete-time Kalman filter, are presented in section 5.2 and 5.3 respectively. The chapter concludes with considerations regarding the tuning process of the filters in section 5.4.

5.1 Peak-to-Peak Moving Average Filter

To obtain a reasonable angular deflection measurement from the high-g and high-frequency dynamics along the z-axis of the accelerometer, three approaches come to mind. The first approach is a mechanical damper, such as those used in UAVs, to smooth out the high-amplitude, high-frequency vibrations, caused by the rotors, which disturb the accelerometer measurement [50]. Silicon foam materials of different shapes or stacked Sorbothane© sheets can be used as one of the more cost-effective mechanical dampers [50]. Simulating a foam material as a PT2 element can prove difficult, since no information about the relative damping, ζ, nor the natural frequency, ω_n, is available. The other two approaches are digital; this work attempts to find an appropriate digital solution. It is important to note that no remotely satisfying estimation result could be achieved by only using the main filter concepts. For the prefilter, two solutions were considered and will be qualitatively compared in section 5.1.2:

First, a discrete-time second order Low Pass filter, which can be described by the this continuous-time transfer function in the frequency domain:

$$G(s) = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2} \; with \; \zeta = \frac{\sqrt{2}}{2}. \qquad \textbf{Eq. (5–1)}$$

The natural frequency or cut-off frequency can be adjusted to smooth out the undesired frequencies. However, there is a trade-off between the damping performance and the increasing phase lag.

The second digital approach is a moving average filter, which essentially represents a time-discrete first order LPF. For this work a standard moving average filter was modified to filter out the high-g, high frequency dynamics of the RCS mounted on the inverse pendulum. Since it will take advantage of the regularly recurring, alternating, uniform peaks in acceleration (see Fig. 4-16), it will be dubbed Peak-to-Peak Moving Average filter (P2PMAF). As with any moving average filter, the number of samples to be averaged must first be determined. This number shall be defined as the subset size, N_{P2P}. For the accelerometer measurements along the z-axis, $a_{k,meas,A,z}$, the subset of measurements can therefore be specified as:

$$\mathbf{v}_{\mathrm{P2P}} = \{a_{k-(N_{P2P}-1),meas,A,z}, a_{k-(N_{P2P}-2),meas,A,z}, \dots, a_{k-1,meas,A,z}, a_{k,meas,A,z}\} \qquad \textbf{Eq. (5–2)}$$

$$N_{P2P} \in \mathbb{N} \; and \; \mathbf{v}_{\mathrm{P2P}} \in \mathbb{N}^{N_{P2P}\times 1}.$$

Now the maximum and minimum values of the subset $\mathbf{v}_{\mathrm{P2P}}$ is calculated and averaged as:

$$a_{k,P2P,z} = \frac{1}{2}(\max(\mathbf{v}_{P2P}) + \min(\mathbf{v}_{P2P})). \qquad \textbf{Eq. (5–3)}$$

The result is an unsteady but smoothed measurement, $a_{k,P2P,z}$, depicted in Fig. 5-1, which can be used together with the x-axis measurement to calculate the angle around the y-axis. This is performed by calculating the arc tangent of the ratio between the two accelerations and converting the result from radians to degrees:

$$\theta_{k,prefiltered} = \arctan\left(\frac{a_{k,P2P,z}}{a_{k,meas,A,x}}\right)\frac{180}{\pi}. \qquad \textbf{Eq. (5–4)}$$

5.1.1 Outlier Analysis

In Fig. 5-1, the three outliers in $a_{k,P2P,z}$ are noticeable:

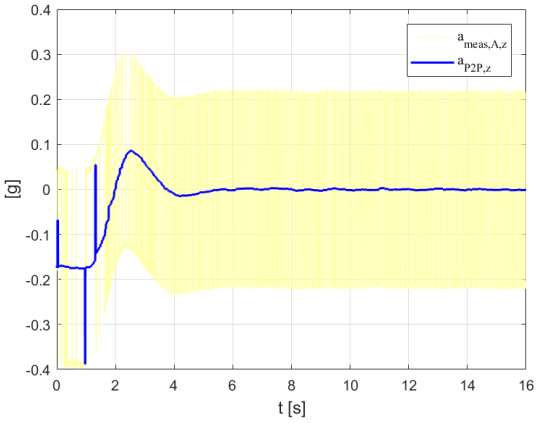


Fig. 5-1: z-axis accelerometer measurement data, raw and filtered by the P2PMAF for a subset size $N_{P2P} = 300$.

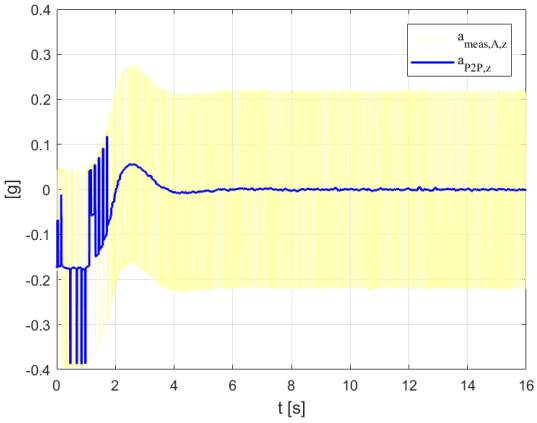


Fig. 5-2: z-axis accelerometer measurement data, raw and filtered by the P2PMAF for a subset size $N_{P2P} = 100$.

The first is equivalent to the rise time, t_r, of a Low Pass filter and is more of a simulation issue, since for the start of the simulation $a_{0,meas,A,Z} = 0$. This minimum is carried through the first subset and defines the rise time for the NP2PMAF,

$$t_{r,P2P} = N_{P2P}t_s = \Delta t_{P2P}, \qquad \textbf{Eq. (5–5)}$$

with the sensor sample time,

$$t_s = \frac{1}{f_{sensor}}.$$

The subsequent outliers result from the heavy reliance of the filter on alternating accelerometer measurements on the z-axis. If exclusively one actuator is active over a period exceeding the subset time window, Δt_{P2P}, an outlier will occur for as long as the second actuator is not active. Outliers will negatively impact the state estimation of the main filter, which is explained in 5.4.3. Fig. 5-2 shows that decreasing the subset size will lead to an increase in the number of outliers. The Fig. 5-3 graph confirms this relationship between the number of outliers, $N_{Outliers}$, and the subset size, N_{P2P}.

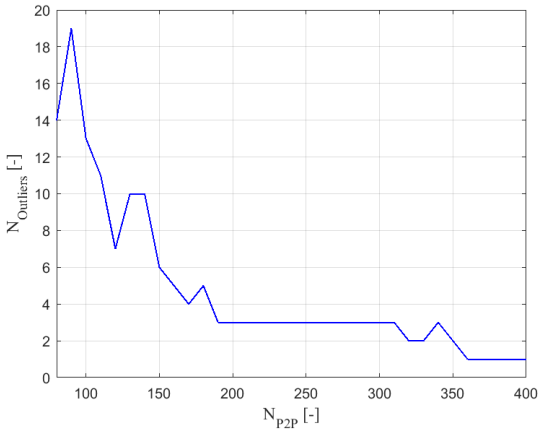


Fig. 5-3: An increase in N_{P2P} leads to fewer outliers over the course of the reference trajectory.

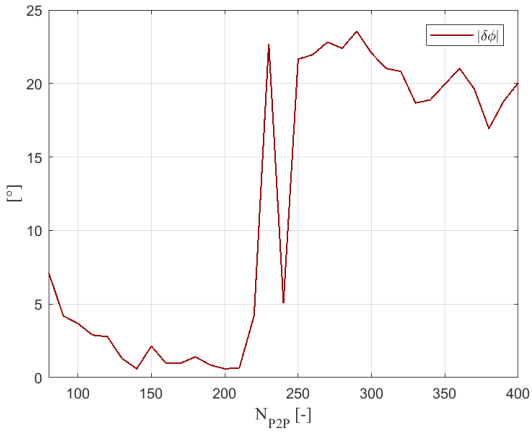


Fig. 5-4: An increase in N_{P2P} leads to a higher absolute phase lag, $|\delta\phi|$, between the true trajectory, $\theta(t)$, and the prefiltered accelerometer data, $\theta_{k,prefiltered}$.

However, as for any moving average / first order Low Pass filter, there is a tradeoff between the smoothing properties and the resulting phase lag. A higher N_{P2P} may cancel out all outliers, except the one caused by the rise time, but at the price of the growing phase lag $|\delta\phi|$ between the true trajectory, $\theta(t)$, and the prefiltered accelerometer data, $\theta_{k,prefiltered}$. This effect is visualized in Fig. 5-4. The phase lag was computed by using the Fast Fourie transform [51, 52] of the smoothed data. The smoothing was performed by the MATLAB function `fit(x,y,'smoothingspline','SmoothingParam',0.07)` [53]. It is evident that there ought to be an optimal subset size for different applications, which will be discussed in 5.4.3.

For the next section, the performance of a second order LPF and the P2PMAF is compared with the aid of an example.

5.1.2 Comparison Between Second Order LPF and P2PMAF

Since no suitable cut-off frequency, f_{co}, for the second order LPF that would allow for a successful stabilization of the pendulum model was found, an example dataset will help to compare the two filters.

The example dataset, θ_{ex}, is distorted, θ_{exd}, to achieve the same regularly recurring, alternating, uniform peaks as those in the accelerometer measurements in the main simulation (see Fig. 5-5). The ability of the two filters to effectively filter out the peaks will be compared by using a standard filter comparison method, the $RMSE$, between the filtered output and the undistorted input, and by the phase lag as described in the previous section. The filter output of the second order LPF and the P2PMAF on $RMSE$-optimised cutoff frequency, f_{co}, and subset size, N_{P2P}, is shown in Fig. 5-6. Results are documented in Table 5-1.

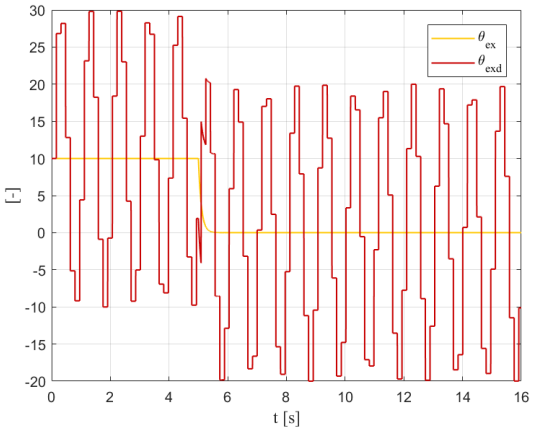


Fig. 5-5: Example dataset, θ_{ex}, and distorted dataset, θ_{exd}.

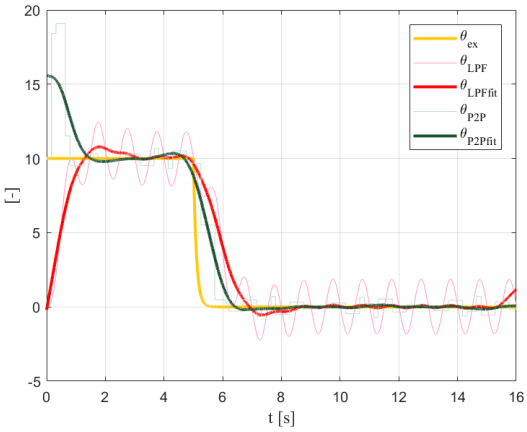


Fig. 5-6: Second order LPF output and P2PMA output in optimal configuration with their respective fits.

Further, in Fig. 5-6, the rise times, $t_{r,P2P}$ and $t_{r,LPF}$, of both filters can be observed during the first two seconds of the simulation. In this example the rise times are more prominent then in the main simulation, since the tuning parameters are set to "more extreme" values. "More Extreme" in the sense of a higher N_{P2P} and a lower f_{co}.

Table 5-1: Comparison of the two filters with the RMSE-optimized tuning parameters. The $RMSE$ is calculated between the respective filter output and the undistorted example trajectory, θ_{ex}.

Filter	**Tuning**	**Root-Mean-Square-Error**	**Phase Lag**
2. Order LPF	$f_{co} = 0.31$ [HZ]	$RMSE = 2.585$ [−]	$\lvert\delta\phi\rvert = 13.112$ [°]
P2PMAF	$N_{P2P} = 800$ [−]	$RMSE = 2.012$ [−]	$\lvert\delta\phi\rvert = 0.763$ [°]

Because of the smaller $RMSE$ and phase lag, the P2PMAF is selected to prefilter the z-axis accelerometer measurements.

However, it is important to reiterate that no subset size could be found that allows sole reliance on the accelerometer data prefiltered by the P2PMAF, $\theta_{k,prefiltered}$, for controlling the attitude. It can be argued that the inverse pendulum without any damping, see 4.1, is too sensitive to even small phase delays or to the resulting outliers.

In the following sections, the main filter designs will be elaborated.

5.2 The Complementary Filter

After prefiltering the z-axis accelerometer measurements with the P2PMAF and subsequently calculating the angle with the x-axis measurement, a standard Complementary filter (CF) can be used to obtain a state estimation. In this chapter, the mathematical fundamentals of the CF and the implementation of the algorithm in the Simulink model will be elaborated upon. A flow chart of the implemented algorithm is given at the end of this section in Fig. 5-7.

5.2.1 Theoretical Background

The Complementary filter is known as a simple and resource-effective filtering algorithm, which is often used for state estimation with low-cost IMUs in unmanned aerial vehicles [19, 54]. In principle, it implements a time-discrete Low Pass filter on the accelerometer measurements to filter out the high frequency noise and a time-discrete High Pass filter on the gyroscope data to filter out the low-frequency drift caused by integration of bias in the angular velocity. It proceeds to add both values together for the state estimate.

The general equation for the Complementary filter can be formulated as follows:

$$\hat{\boldsymbol{x}}_k = K_c\left(\hat{\boldsymbol{x}}_{k-1} + \dot{\boldsymbol{x}}_{k,meas,G}\Delta t\right) + (1 - K_c)\boldsymbol{x}_{k,meas,A} \qquad \textbf{Eq. (5–6)}$$

$$0 < K_C \leq 1,$$

where K_C is defined as the Complementary filter gain and Δt is the interval time between CF iterations. The gain can be changed to suit the needs of the application. A high K_C can be interpreted as a high gyroscope sensitivity, usually leading to a less volatile output. A low K_C can be interpreted as a high accelerometer sensitivity, leading to less drift in the output in the long term [54]. For $K_C = 1$, the accelerometer input is ignored, and the algorithm would only consider the gyroscope input over time. The drift resulting from the integration would then equal the constant residual bias in the gyroscope, $\dot{\boldsymbol{x}}_{RB,G}\ °/s$, which cannot be eliminated through calibration Fig. 4-11.

5.2.2 Implementation in the Simulation

For the 1-DoF case, Eq. (5–6) can be formulated as follow:

$$\hat{\theta}_k = K_c * \left(\hat{\theta}_{k-1} + \dot{\theta}_{k,meas,G} * \Delta t\right) + (1 - K_c) * \theta_{k,meas,\mathrm{A}}, \qquad \textbf{Eq. (5–7)}$$

with the time interval between iterations being

$$\Delta t = \frac{1}{f_{sensor}} = t_s, \qquad \textbf{Eq. (5–8)}$$

and the estimated angle around the y-axis of the inverse pendulum, $\hat{\theta}_k$.

The algorithm in Eq. (5–7) was implemented by using the `MATLAB Function` block and a `Delay` element. The corresponding Simulink model is depicted in the appendix Fig. D-13. Since it is assumed that the initial position of the pendulum is unknown, the initial state estimate , θ_0, will be set to zero. This will lead to a filter rise time, which the buffer time at the start of the chosen commanded trajectory accounts for.

Before explaining the tuning considerations, the discrete Kalman filter, a more sophisticated estimation algorithm, is introduced.

To sum up the complementary filter algorithm, a flow chart is depicted in Fig. 5-7.

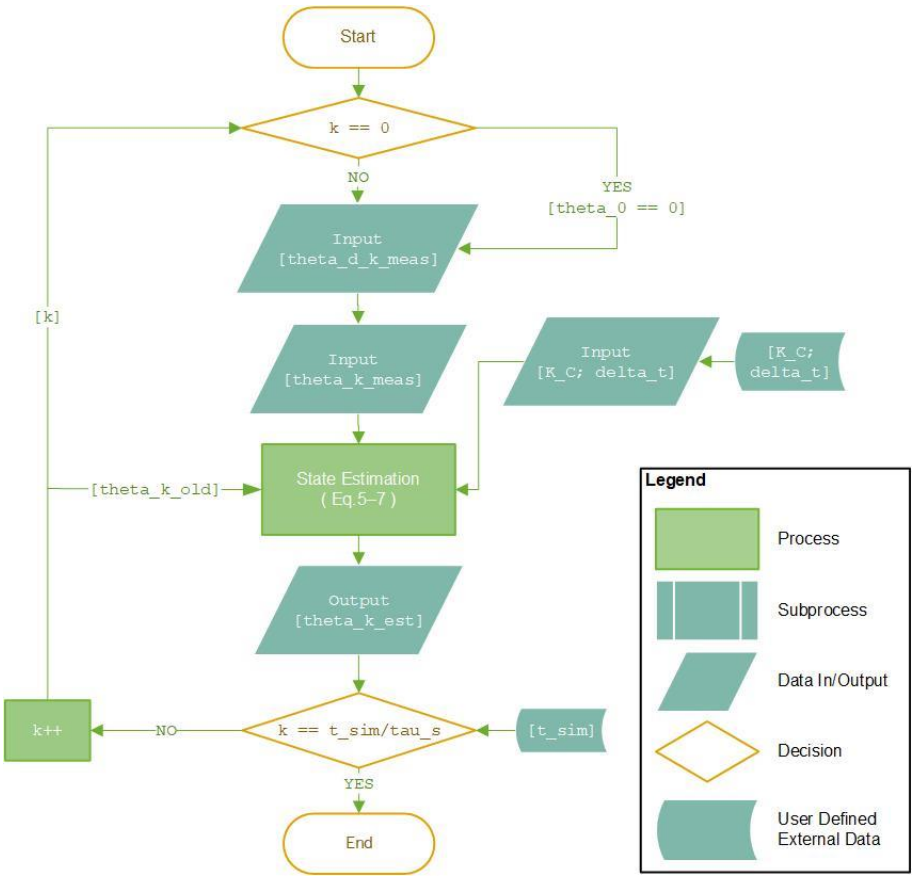


Fig. 5-7: Complementary filter flow chart.

5.3 The Discrete Kalman Filter

The Kalman filter (KF) is the second main filter concept to be designed and evaluated. It forms the basis for most state estimation algorithms used in navigation and attitude determination systems [12]. The KF is a Bayesian estimator able to obtain optimal estimates using knowledge of the deterministic and statistical properties of the system parameters and measurements [12]. This chapter explains the underlying algorithms and how they can be simplified for implementation in the model. Then, an extension to the Kalman filter is introduced to improve the results. A flow chart of the implemented algorithm is given at the end of this section in Fig. 5-11.

5.3.1 Theoretical Background

Since the filter will handle time discrete data from the sensors, the discrete version of the KF algorithm is used.

The discrete-time Kalman filter algorithm essentially consists of two phases: The system propagation or prediction phase and the measurement update or correction phase, which are illustrated in Fig. 5-8. As the name suggests, the prediction phase predicts the state vector estimate and error covariance matrix when transitioning between timesteps [12]. In the correction phase, the state vector estimate and error covariance are updated to incorporate the new measurement information via the Kalman gain matrix, $\boldsymbol{K}_K$. This matrix optimally weighs the correction to the state vector

according to the uncertainty of the current state estimates and the level of noise in the measurements [12].

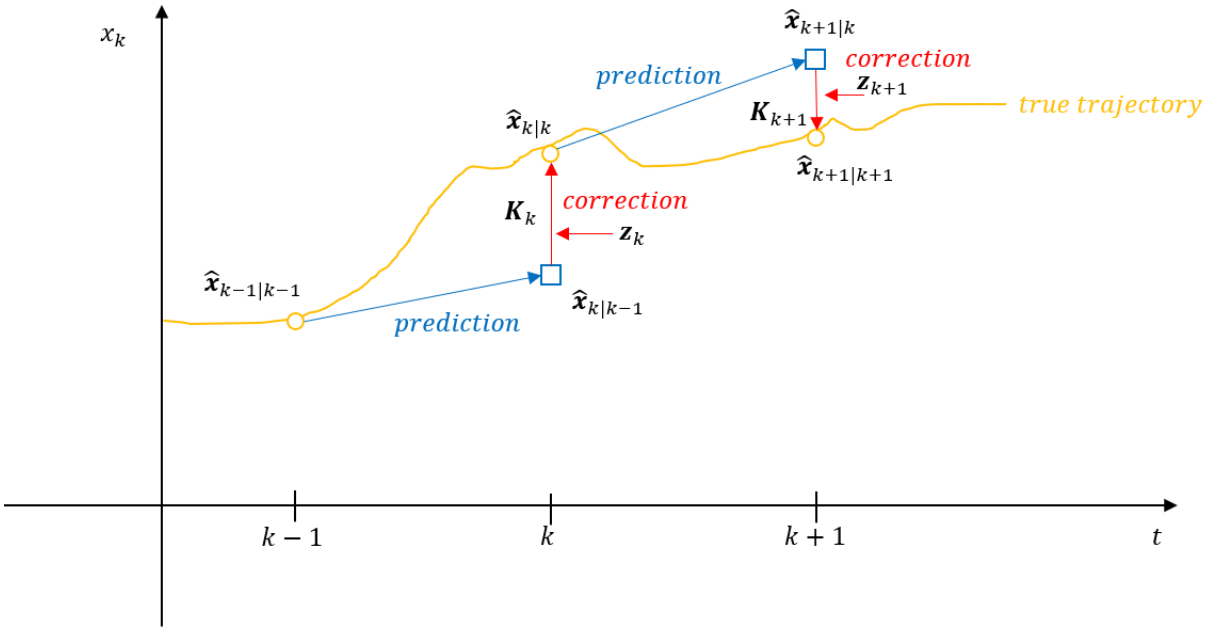


Fig. 5-8: The two phases of the Kalman filter are depicted for two iterations. Adapted from [55].

The two overarching phases of the discrete-time Kalman filter can be further broken down into 10 steps. The first 4 steps are part of the prediction phase, while steps 5 to 10 form the correction phase. The following procedure is completely taken from [12] and simplified to fit into the scope:

Table 5-2: The 10 steps of the discrete-time Kalman filter algorithm taken from [12].

ID	Description	Phase
K01	Calculate the transition matrix $\boldsymbol{\Phi}_{k-1}$	Prediction
K02	Calculate the system / process noise covariance matrix $\boldsymbol{Q}_{k-1}$	Prediction
K03	Propagate the state vector estimate from $\hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{+}$ and $\hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{-}$	Prediction
K04	Propagate the error covariance matrix from $\boldsymbol{P}_{k-1}^{+}$ to $\boldsymbol{P}_{k-1}^{-}$	Prediction
K05	Calculate the measurement matrix $\boldsymbol{H}_k$	Correction
K06	Calculate the measurement noise covariance matrix $\boldsymbol{R}_k$	Correction
K07	Calculate the Kalman gain matrix $\boldsymbol{K}_K$	Correction
K08	Formulate the measurement $\boldsymbol{z}_k$	Correction
K09	Update the state vector estimate from $\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-}$ and $\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{+}$	Correction
K10	Update the error covariance matrix from $\boldsymbol{P}_k^{-}$ to $\boldsymbol{P}_k^{+}$	Correction

The discrete transition matrix $\boldsymbol{\Phi}_{k-1}$ is different for every Kalman filter application and is derived from the linearized system model. In most cases, it is a function of the time interval, τ_s, between Kalman filter iterations.

Step K02 calculates the system noise covariance matrix. It characterizes the growing uncertainty in the system model with each iteration. However, for the standard discrete-time KF, the system noise covariance matrix $\boldsymbol{Q}$ and the measurement noise covariance matrix $\boldsymbol{R}_k$ remain constant and are never updated ($\boldsymbol{R}_k = const. = \boldsymbol{R}$) . Thus, the steps K02 and K06 are omitted. Step K03 estimates the propagation of the state vector through time using:

$$\hat{\boldsymbol{x}}_k^- = \boldsymbol{\Phi}_{\mathrm{k}-1}\hat{\boldsymbol{x}}_{k-1}^+. \qquad \textbf{Eq. (5–9)}$$

Step K04 completes the prediction phase by propagating the corresponding error covariance matrix:

$$\boldsymbol{P}_k^- = \boldsymbol{\Phi}_{k-1}\boldsymbol{P}_{k-1}^+\boldsymbol{\Phi}_{k-1}^T + \boldsymbol{Q}. \qquad \textbf{Eq. (5–10)}$$

The correction phase starts with the calculation of the measurement matrix $\boldsymbol{H}_k$, which defines how the measurement vector varies with the state vector. In a standard Kalman filter, each measurement is assumed to be a linear function of the state vector. Therefore,

$$h(\boldsymbol{x}_k, t_k) = \boldsymbol{H}_k\boldsymbol{x}_k. \qquad \textbf{Eq. (5–11)}$$

However, for the 1-DoF case the measurement matrix is simplified to $\boldsymbol{H}_K = 1$, and thus, the measurement z_k, from step K08, simplifies to the true 1D position angle, θ_{k}, plus the measurement noise, w:

$$z_k = \theta_{\mathrm{k}} + w. \qquad \textbf{Eq. (5–12)}$$

In step K07 the Kalman gain matrix is calculated. It depends on the error covariance matrices of both the true measurement vector, z_k, and that predicted from the estimates, $\boldsymbol{H}_{\mathrm{k}}\hat{\boldsymbol{x}}_k^-$. To provide an intuitive explanation, the Kalman gain can be interpreted as a ratio of the uncertainty in the model, with $\boldsymbol{P}_k^- = f(\boldsymbol{Q})$**,** to the uncertainty in the measurement [55]:

$$\boldsymbol{K}_K = \frac{uncertainty\ in\ the\ propagation}{uncertainty\ in\ the\ measurement} = \frac{\boldsymbol{P}_k^-\boldsymbol{H}_k^T}{\boldsymbol{H}_k\boldsymbol{P}_k^-\boldsymbol{H}_k^T+\boldsymbol{R}}. \qquad \textbf{Eq. (5–13)}$$

The Kalman gain will be used to weigh the predicted a-priori state estimate, $\hat{\boldsymbol{x}}_k^-$, to the measurement innovation, $\boldsymbol{y}_k$, to produce the a-posteriori state estimate, $\hat{\boldsymbol{x}}_k^+$, in step K09:

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k\boldsymbol{y}_k \qquad \textbf{Eq. (5–14)}$$

with,

$$\boldsymbol{y}_k = z_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_k^-. \qquad \textbf{Eq. (5–15)}$$

With this step, the state vector estimate is corrected by the measurement update. Correspondingly, the error covariance matrix is updated in K10 with:

$$\boldsymbol{P}_k^+ = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{P}_k^-. \qquad \textbf{Eq. (5–16)}$$

Since the updated state vector estimate is based on more information, the updated state uncertainties are smaller than before. The next chapter will apply the principles

to the considered scenario established in 3.1.2, which will vastly reduce the complexity of the algorithm.

5.3.2 Implementation in the Simulation

For the 1-DoF case, only the attitude around the inertial frame y-axis of the inverse pendulum needs to be estimated. This reduces the state vector $\boldsymbol{x}_k$ to the scalar angle θ_k (see 3.1.2). Since only θ_k needs to be estimated, the Eq. (5–9) from step K03 simplifies to the kinematic equation:

$$\hat{\theta}_k^- = \Phi_{\mathrm{k}-1}\hat{\theta}_{k-1}^+ + \tau_s\dot{\theta}_{k-1,meas,G} \qquad \textbf{Eq. (5–17)}$$

with,

$$\Phi_{\mathrm{k}-1} = 1 \; and \; \tau_s = \frac{1}{f_{sensor}} = t_s. \qquad \textbf{Eq. (5–18)}$$

Recalling that the steps K02 and K06 are omitted, the algorithm in the Simulink model effectively starts with the calculation of the a-priori estimate Eq. (5–9). Now, only the error covariance must be calculated to complete the prediction phase:

$$P_k^- = P_{k-1}^+ + Q. \qquad \textbf{Eq. (5–19)}$$

The error covariance matrix, $\boldsymbol{P}_k^-$, and the system noise covariance matrix, $\boldsymbol{Q}$, are reduced to the scalar error variance, P_k^-, and system noise variance, Q. As stated in 5.3.1, the measurement matrix, $\boldsymbol{H}_{\mathrm{k}}$, will be omitted. On this basis, the correction phase starts with step K07 by calculating the scalar Kalman gain, K_{k}.

$$K_K = \frac{P_k^-}{P_k^- + R} \qquad \textbf{Eq. (5–20)}$$

In accordance with Table 5-2, the a-posteriori estimation of the attitude angle can be calculated:

$$\hat{\theta}_k = \hat{\theta}_k^+ = \hat{\theta}_k^- + K_k y_k, \qquad \textbf{Eq. (5–21)}$$

with the measurement innovation,

$$y_k = \theta_{k,meas,A} - \hat{\theta}_k^-. \qquad \textbf{Eq. (5–22)}$$

Finally, the a-posteriori error variance is computed with the equation:

$$P_k^+ = (1 - K_k)P_k^-. \qquad \textbf{Eq. (5–23)}$$

A summary of the algorithm extended by an innovation-based adaptive estimator is depicted in Fig. 5-11.

In Simulink, the algorithm can be implemented by using a `MATLAB Function` block and the use of `Delay` elements (see Fig. D-14). The sample rate of the whole subsystem and all included blocks is set to the sample rate of the sensor, f_{sensor}. Alternatively, the `Delay` elements can be replaced by using `persistent` variables in the `MATLAB Function` block (see [56]). This, however, will require an additional if-loop in the function for the initialization variables, which may arguably deteriorate the transparency and readability of the model.

For the first iteration of the loop, $k = 0$, the values for $\hat{\theta}_{k-1}^+$, $\dot{\theta}_{k-1,meas,G}$ and P_{k-1}^+ must be initialized. Similiar to the measurement variance R and the process variance Q, the

initialization can be performed through a customized mask. A guide on how to use the mask is given in the appendix Chapter B. An approach on how choose the initial variables for the KF is explained in 5.4.2.

Of course, there is also the option to use the `Kalman Filter` block, referred to as Simulink Kalman Filter (SKF), which is part of the Simulink `Control System Toolbox` [57]. However, in this thesis, a customized KF was designed to allow enhancement by applying an Innovation-based Adaptive Estimator (IAE), which will be discussed in the next section. Fig. 5-9 plots the difference, $\delta\theta_{SKFKF}$, between the angle estimated by the implemented KF, $\hat{\theta}_k$, and the prefabricated SKF block, $\hat{\theta}_{k,SKF}$. Both filters are initialized with the same values for Q, P_0, R and θ_0.

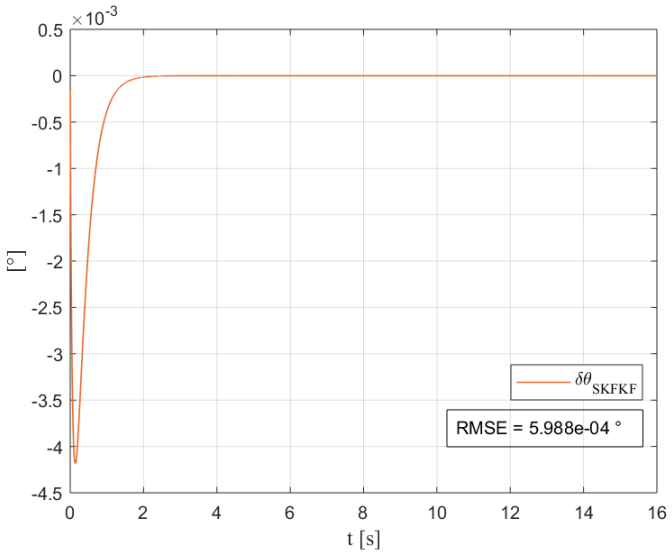


Fig. 5-9: The graph shows the difference, $\delta\theta_{SKFKF}$, in degrees over time as well as the $RMSE$ between the estimated angle, $\hat{\theta}_k$, by the implemented Kalman filter and the Simulink Kalman filter $\hat{\theta}_{k,SKF}$.

It can be recognized that the error between the two estimation spikes at the beginning and then quickly converges to zero. For this experiment the trajectory, $\theta_{com}(T_{mode} = 3)$, suggested in 4.2, was chosen. Therefore, this spike is not due to the commanded step input, which occurs at $t = 1\,\mathrm{s}$. Upon inspection of the SKF block, it was found that for step K03 the current gyroscope measurement input, $\dot{\theta}_{k,meas,G}$, is used for the system propagation, instead of $\dot{\theta}_{k-1,meas,G}$ (see Eq. (5–17)). Therefore, the SKF does not initialize $\dot{\theta}_{k,meas,G}$ with a value defined by the user. This is the only difference found between the two algorithms. The relatively small spike at the beginning of the trajectory with a magnitude of $\delta\theta_{SKFKF} = 0.0042\,°$ has no detectable influence on the performance.

5.3.3 Innovation-Based Adaptive Estimation Extension

For most applications, the KF's system noise covariance matrix, $\boldsymbol{Q}$, and measurement noise covariance matrix, $\boldsymbol{R}$, are constant values, determined during the development phase through laboratory measurements of the system, simulation and trials [12]. However, there are cases where this is not possible. Vibrations in highly dynamic environments cannot always be simulated or tested. Additionally, the sensor

performance may deteriorate over several years of usage, which is also difficult to simulate. In other cases, the optimum KF tuning might vary over time as the respective context changes. For example, a GNSS navigation filter in a mobile device that may be stationary, on a walking pedestrian, or in a car, would require a different system noise model in each case [12]. For these applications an adaptive Kalman filter (AKF) may be used to estimate $\boldsymbol{R}$ and $\boldsymbol{Q}$ as it operates. One possible approach is the Innovation-based Adaptive Estimation, which is detailed in [58, 59]. Essentially, the IAE updates $\boldsymbol{R}$ and $\boldsymbol{Q}$ based on the relevant measurement innovation statistics.

For this work, a vastly simplified version of the IAE depicted in [58] was implemented. It will help to address the issues arising from the outliers caused by the P2PMAF described in Chapter 5.1. Similarly, to the previously described use cases for adaptive estimation, the exact point in time and in which magnitude the outliers occur is impossible to know beforehand. Therefore, it is difficult to involve them in the calculation of the measurement variance R. In the context of this work, the measurement innovation, y_k, is the difference between $\theta_{k,meas,A}$, the angle prefiltered and calculated from the accelerometer measurement, and the a-priori estimate, $\hat{\theta}_k^-$.

$$y_k = \theta_{k,meas,A} - \hat{\theta}_k^- \qquad \textbf{Eq. (5–24)}$$

Based on the absolute value of the measurement innovation the decision is made on whether there is an outlier or not. For y_k to be classified as outlier-based, it must be greater than a chosen threshold, T_{IAE}. For an outlier-based y_k, the uncertainty in the measurement, effectively R, will be set to infinity and, consequently, the Kalman gain will be reduced to zero.

$$K_K: \begin{cases} 0, & for\ |y_k| > T_{IAE} \\ K_K, for\ |y_k| \leq T_{IAE} \end{cases} \qquad \textbf{Eq. (5–25)}$$

The threshold, $T_{IAE} = 3\,°$, was chosen to exceed all non-outlier-based measurement innovations. For the set time interval, τ_s (see Eq. (5–18)), a $y_k > 3\,°$ would be equal to angular acceleration of $> 3000\,°/s$ for a sensor sample rate of $1000\,\mathrm{Hz}$, which classifies the measurement as an outlier. Outliers are visualized in Fig. 5-10.

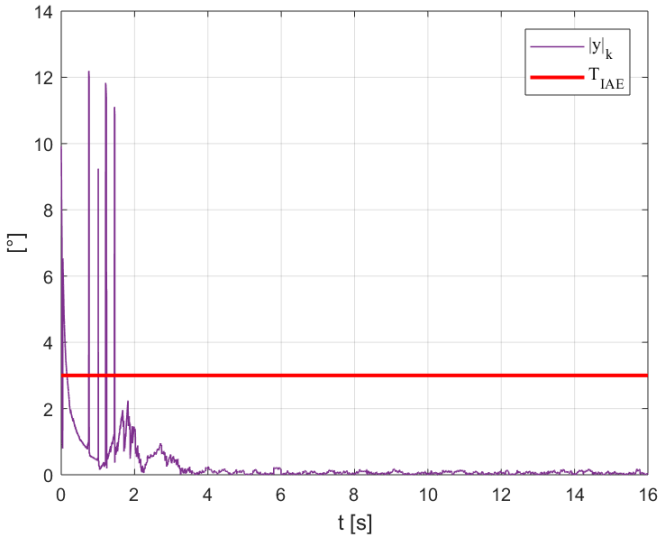


Fig. 5-10: Absolute measurement innovation over the course of the experimental trajectory, $T_{mode} = 3$, with a $N_{P2P} = 200$ and reference values.

The figure above shows that outliers surpass the threshold on 4 occasions, therefore setting the Kalman gain to zero. For the first few iterations the $|y_k|$ values are relatively high. This is due to the rise time of the KF. Therefore, a small buffer of $0.5\,s$ was implemented, in which the IAE will not be activated. This buffer value may be changed if other trajectories are chosen.

To conclude this chapter, a data flow chart of the implemented algorithms is depicted in Fig. 5-11.

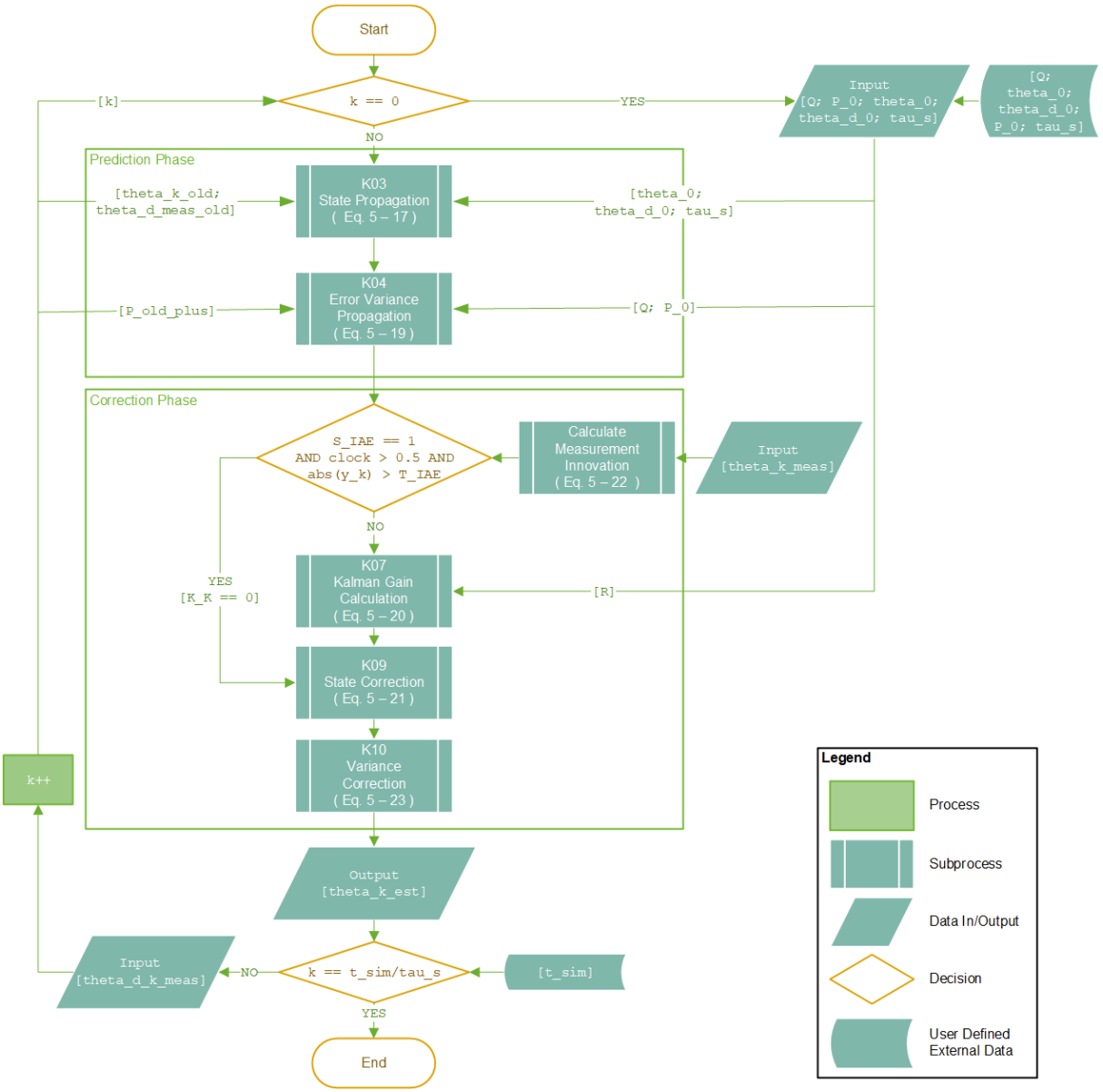


Fig. 5-11: Flow chart of the implemented discrete-time KF with an IAE extension referencing the steps from Table 5-2. The syntax of the `KalmanFilterEquations` block in Fig. D-14 was used.

5.4 Filter Tuning

After introducing the prefilter and the two main filter algorithms, the topic of this chapter will be to explain best-practice approaches for tuning the respective filter. The

knowledge for this procedure was obtained from [12], during the development phase and while performing the preliminary test series, detailed in Chapter 6.2.

5.4.1 Complementary Filter Tuning Considerations

Tuning the CF can be achieved by optimizing the Complementary filter gain, K_C, which may vary depending on the measurement characteristic and performance requirements. Since the accelerometer measurements are prefiltered by the P2PMAF, the calculated angle will be less volatile, but will exhibit a certain lag depending on the subset size, N_{P2P}. Hence, for this application, the K_C value must be above a certain threshold for the system to be able to stabilize itself and not hit the ends top of the test stand. Simply iterating through increasing gain values while keeping all other simulation parameters at the constant reference variables listed in 6.1., suggests a local optimum at a root-mean-square-error (see 6.2.1) of $RMSE = 1.78\,°$ for a $K_C = 0.9987$. The $RMSE(K_C)$ plot is depicted in Fig. 5-12 and the true trajectory is shown in Fig. 5-13, alongside two other exemplary true trajectories and the commanded trajectory.

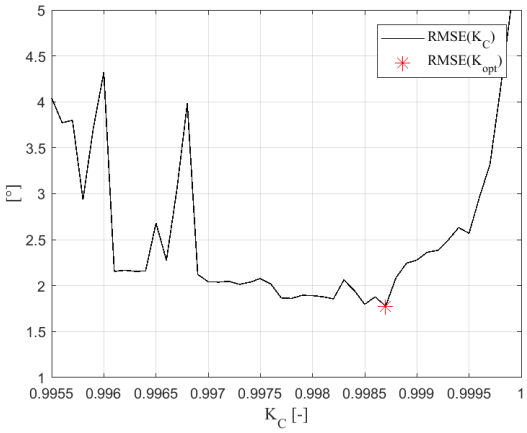


Fig. 5-12: $RMSE(K_C)$ plot for reference variables with local optimum at $K_C = 0.9987$.

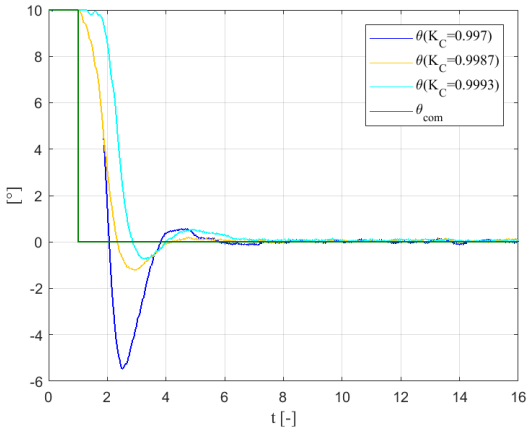


Fig. 5-13: Exemplary true trajectories for three different K_C values.

The trajectories in Fig. 5-13 suggest that a decrease in K_C will lead to a more aggressive stabilization process with larger overshoot but faster zero-crossing, and an increase in the gain will lead to the opposite. This must be considered when facing changing circumstances due to hardware or software design decisions.

5.4.2 Kalman Filter Tuning Approach

For tuning the Kalman filter as implemented in the previous chapter, the system variance, Q, the measurement noise variance, R, the initial state values, θ_0 and $\dot{\theta}_0$, and the initial error variance, P_0, must be defined by the developer or user. Both initial states are set to zero assuming that the initial state of the system is unknown. This results in an initial error variance of

$$P_0 = (\theta_{ES})^2. \qquad \textbf{Eq. (5–26)}$$

Suggesting that the initial error or uncertainty in the system is most likely not larger than the maximum deflection of the pendulum confined by the end stop. Setting the

initial state values to zero will result in a rise time which decreases with increasing P_0 or decreasing R. However, minimizing the rise time, and therefore risking performance losses, is not necessary, since the initial buffer time of one second can account for it.

The system variance and the measurement variance are closely related to their respective sources. In the state prediction, the used kinematic model is based on the gyroscope measurements and the measurement innovation is based on the prefiltered accelerometer output. Therefore, both variances are exposed to the respective sensor noise. In several studies concerning the characterization of the MPU 6050, the noise was classified as white gaussian in nature [21, 46]. This stochastic error process can be represented by its standard deviation, σ. The standard deviation for the sensors established in 4.5 can now be used for giving an appropriate estimation of the two variances:

$$Q = {\sigma_{G,y}}^2 \qquad \textbf{Eq. (5–27)}$$

$$R = \left(\left(\operatorname{atan}\frac{\sigma_{A,z}}{\sigma_{A,x}}\right)\frac{180}{\pi}\right)^2 \qquad \textbf{Eq. (5–28)}$$

It should be noted, that for R, the respective standard deviations, $\sigma_{A,z}$ and $\sigma_{A,x}$, are given in the unit $\mathrm{m/s^2}$ and therefore must be converted from rad into ° by using Eq. (5–4) established in Section 5.1. This forms the basis for tuning the Kalman filter algorithm. The computed variances will serve as the reference values for further tests in 6.1.1.

To conclude the chapter, some qualitative best-practice tuning guidelines are provided, which will be expanded upon in Section 7.1. This shall justify why Q is chosen to further optimize the tuning given a certain error profile.

If the result after tuning is not satisfactory, [12] suggests optimizing one of the parameters by fixing the others. Results from the preliminary test series in 6.2 suggest that by varying Q, unwanted effects like a constant offset or drift of the true trajectory can be accounted for. Fig. 5-14 depicts the course of the true trajectory with all simulation parameters set to the reference values, documented in Chapter 6.1, except the residual bias, $\omega_{RB,G,y}$, which is increased by some arbitrary factor C (for Fig. 5-14 and Fig. 5-15, $C = 20$). An offset of roughly 0.4 ° can be identified, which results from a faulty estimation. This loss in estimation performance can be accounted for by increasing the variance by the same factor C resulting in

$$Q = C{\sigma_{G,y}}^2. \qquad \textbf{Eq. (5–29)}$$

However, this will result in a lag in the estimation since the Kalman gain, K_K, will now converge towards a higher value, which gives more weight to the measurement innovation, which has LPF-properties as established in 5.1. This circumstance is shown in Fig. 5-15.

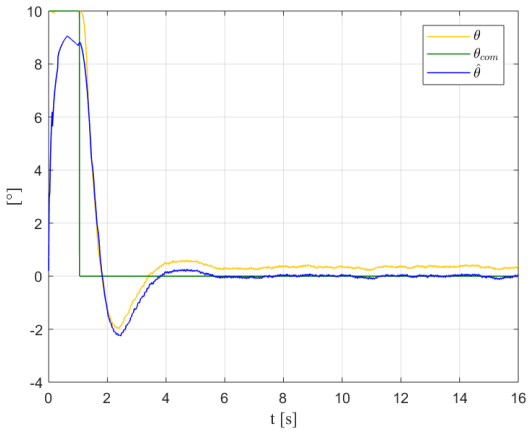


Fig. 5-14: Constant offset in true trajectory due to twentyfold increase in $\omega_{RB,G,y}$, which results in a faulty estimation.

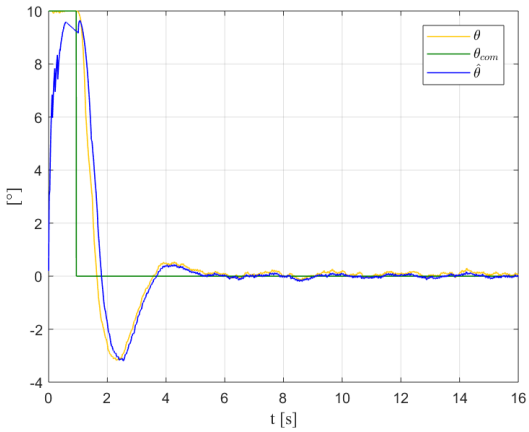


Fig. 5-15: Offset in true trajectory due to 20-times increase in $\omega_{RB,G,y}$, is accounted for by setting $C = 20$.

In general, decreasing the ratio Q/R leads to higher lag in the estimation, which may result in a larger overshoot, θ_{OS}. While increasing the ratio may decrease the aggressiveness of the stabilization process to a point where the true trajectory does not possess steady state accuracy anymore, for the given timeframe.

Thus, based on Fig. 5-14 and Fig. 5-15, Q should be multiplied by a tuning constant C, to obtain satisfactory performance results.

5.4.3 Moving Average Subset-Size Considerations

In Section 5.1.1 it was established that there is an optimum subset size, N_{P2P}, to reduce the $RMSE$. A too low subset size will increase the number of outliers and a too high subset size will increase the phase lag. Evidently, outliers negatively impact the estimation accuracy of the main filter and even increase the phase lag if the outlier count is over a certain threshold as depicted in Fig. 5-4.

It was assumed that the KF and the Complementary filter might possess different optimal N_{P2P}. Therefore, over the course of the preliminary test series, the subset size was varied over a reasonable range to find an optimal configuration for the different filter variants. The results are depicted in the following graphs:

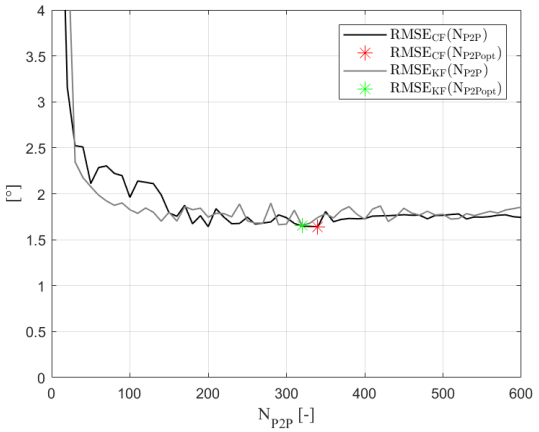


Fig. 5-16: $RMSE(N_{P2P})$-graph for the KF and the CF with their local minima. The variation step size is 10.

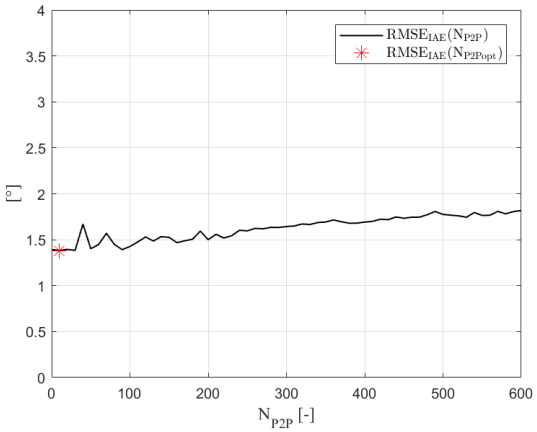


Fig. 5-17: $RMSE(N_{P2P})$-graph for the AKF with its local minima. The variation step size is 10.

Fig. 5-17 reveals that the KF with the IAE extension performs best for exceptionally low or no prefiltering at all. This is an unexpected result, since initially the IAE was introduced for the short comings of the P2PMAF. The results of the optimisation are summed up in Table 5-3.

Table 5-3: $RMSE(N_{P2Popt})$ for the three filter configurations with the respective optimal subset size, N_{P2Popt}.

Filter	**Optimal Subset Size**	**Root-Mean-Square-Error**
Complementary filter	340 [−]	1.640 [°]
Kalman filter	320 [−]	1.656 [°]
AKF	10 [−]	1.365 [°]

Furthermore, Fig. 5-16 and Table 5-3 reveal a comparable graph history for the CF and the KF, when varying the subset size.

Given these results, only the following two filter configuration will be considered for the final evaluation: P2PMAF as the prefilter and the CF as the main filter; KF with IAE extension without prefiltering, with the CF being computational simple and the AKF possessing better performance. This exclusion of other filter configurations will help reduce the complexity of the test series.

With these results, the design part of this work is finished. The next chapter presents a side by side comparison of the performance. There the proposed filter concepts will be evaluated for consistent performance under varying input parameters. Thus, addressing the characterization part of this thesis.

6 Testing and Evaluation

To evaluate the effect of changing input parameters on the filter performance, a wide range of simulation experiments need to be performed. However, the multitude of different variable input parameters complicates a profound analysis of the whole design space. To reduce the complexity of this problem design of experiments (DOE) methods [60] will be combined with Monte Carlo simulation methods [61]. The sample size of such an experiment is referred to as levels, L. With the number of parameters, P, the overall number of simulations runs, N, can be determined for a designed experiment [62].

$$N = L^P \qquad \textbf{Eq. (6–1)}$$

This denomination will be used for designing an experimental setup for the filter evaluation. Nine steps were conducted for designing the simulation experiment:

Table 6-1: The 7 steps of DoE from [63] are marked with the ID RSX. Additional steps introduced for this thesis are marked with ASX. Adopted from [60, 64].

ID	Description	Addressed In
RS1	Recognition and statement of the problem	Chapter 3
AS2	Categorization of all input factors	Section 6.1
RS3	Selection of the response variable(s)	Section 6.2.1
AS4	Conducting and evaluating a wide range of preliminary experiments	Section 6.2.2
RS5	Choice of factors, levels and ranges	Section 6.2.3
RS6	Choice of design	Section 6.2.4
RS7	Conducting the experiment	Section 6.3
RS8	Statistical analysis	Section 6.3
RS9	Drawing conclusions and marking recommendations	Chapter 7

Characterizing the filter performance under changing input parameters represents the problem statement and therefore addresses RS1. The problem statement is broken down into the two research questions, Q2 and Q3, formulated in Chapter 3.

For AS2, all input parameters are classified into three distinct categories and from now on will be referred to as factors. An overview of the entire classification process regarding the design of this simulation experiment is given in Section 6.3 Fig. 6-5.

6.1 Factor Categorization

In [63], factors are categorized into “treatment factors”, which are of primary interest, and “nuisance factors”, which are not the primary focus of the experiment. This distinction will be essential to reduce the complexity of the problem and is performed in Section 6.2.3. To facilitate this process, all factors can also be divided into **internal**, **external** and **error-related** factors. The goal of this distinction is to find interrelations

between internal, external and error-related key factors. Therefore, the following sections will focus on defining the three categories and providing the respective reference values x_{ref}:

6.1.1 Internal Factors

Internal factors are adjustable and can therefore be used to optimize the performance of the control system. This includes controller gains, filter parameters, filter-selection-switches and data rates. A comprehensive list of all internal factors with their respective Simulink subsystem and reference value is presented in Table 6-2.

For the internal factors, the reference values have been proven to satisfy the functional requirements for the chosen external (see Table 3-3) and error-related (see Table 6-4) reference values.

Table 6-2: The table lists all input variables of the simulation which will be labeled as internal, therefore adjustable, factors and their respective Simulink subsystems.

Internal Factor	Reference Value	Description	Simulink Subsystem
f_{sensor}	1000 [Hz]	Sensor sample rate	Sensor
f_{bb}	100 [Hz]	Bang-Bang controller frequency is limited by f_{max}	Controller
u_{bb0}	0.1 [−]	For $u_{PID} < -u_{bb0}$ and $u_{PID} > u_{bb0}$ control output $u_{bb} = 0$	Controller
K_p	0.53 [−]	Proportional gain	Controller
K_i	0.79 [−]	Integrator gain	Controller
K_d	0.09 [−]	Differential gain	Controller
N_{P2P}	340 [−]	P2PMAF subset size for CF taken from 5.4.3	Filter
Q, R, P_0, θ_0		See Tuning Approach in Section 5.4.2	Filter
C	1 [−]	Tuning coefficient Kalman filter	Filter
K_C	0.9987 [−]	Complementary filter gain (see Section 5.4.1)	Filter
f_{co}	250 [Hz]	Built-in LPF cut-off frequency if $S_{LPF} = 1$	Sensor
$\boldsymbol{\omega}_{range}$	$500 \left[\frac{^\circ}{s}\right]$	Selected measurement range	Sensor
$\boldsymbol{a}_{range}$	4 [g]	Selected measurement range	Sensor
s_{LPF}	0 [−]	IMU-integrated LPF switch	Sensor
s_{BC}	1 [−]	Turns the bias compensator ON for $s_{BC} = 1$	Sensor
s_{P2P}	0 [−]	Turn P2PMAF ON for $s_{P2P} = 1$	Filter
s_F	1 [−]	Switch between KF for $s_F = 1$ and CF for $s_F = 0$	Filter
s_{IAE}	1 [−]	Switch between KF for $s_{IAE} = 0$ and AKF for $s_{IAE} = 1$	Filter

6.1.2 External Factors

External factors are determined by the hardware design or the test environment and are therefore, not adjustable. This includes mass of the entire system, computational time delay measured by the implementation-team [32], maximum actuator force and actuator exhaust velocity among other factors. Table 6-3 presents a comprehensive list of all external factors with their respective Simulink subsystem and reference values.

For the external factors, the reference values consist of worst-case estimation by the hardware design team [26], values from the product data sheets [23], functional requirements or assumed environmental conditions.

Table 6-3: The table lists all input variables of the simulation which will be labeled as external, therefore not adjustable, factors and their respective Simulink subsystems.

External Factor	Reference Value	Description	Simulink Subsystem
t_{st}	0.0035 [s]	ON / OFF switching time of the actuator	Actuator
v_{act}	521 $\left[\frac{\mathrm{m}}{\mathrm{s}}\right]$	Average actuator exhaust velocity	Actuator
F_{max}	7 [N]	Maximum force the actuators can produce	Actuator
F_{st}	99 [%]	Achieved force after t_{st} in relation to the maximum Force	Actuator
$t_{actdelay}$	0 [s]	Additional delay on actuator	Actuator
l_A	1 [m]	Actuator lever arm	Pendulum
l_{CM}	0.667 [m]	Assumed center of mass lever arm	Pendulum
θ_{ES}	10 [°]	End stop angle (equal to initial deflection)	Pendulum
m_0	5 [kg]	Point mass of the whole system at $t = 0$	Pendulum
$\omega_{n,G}$	30000 [Hz]	Natural frequency of the gyroscope around y-axes	Sensor
T_E	25 [°C]	Environmental temperature impacts Sensor sensitivity	Sensor
B	16 [bit]	Available bit size influences quantization error together with the measurement range	Sensor
t_{sim}	16 [s]	Simulation time	Trajectory
T_{mode}	3 [−]	Trajectory mode	Trajectory
$t_{PIDdelay}$	0.01 [s]	Computational time delay / latency	Controller
s_m	0 [−]	Activate decaying mass because of gas loss for $s_m = 1$	Pendulum

6.1.3 Error-Related Factors

Error-related factors strictly impair the performance of the control system and are not adjustable. This includes sensor noise, sensor bias and scale factors among other factors. Error-related factors are further divided into deterministic and stochastic errors.

A comprehensive list of all error-related factors with their respective Simulink subsystem and reference value is presented in Table 6-3.

For the error-related factors, calibration errors were extracted from the product datasheets [14, 23] and the standard deviations for the white noise were taken from a sensor characterization study conducted with MPU-6050 [21]. For the reference values, a perfect calibration is assumed therefore setting scale factor, misalignment and static bias to zero. As an exception, the gyroscope residual bias, $\boldsymbol{\omega}_{RB,G}$, is set to $-0.035\ °/s$ to account for requirement F24 (see Section 4.5).

Table 6-4: The table lists all input variables of the simulation which will be labeled as error related, therefore not adjustable and performance impairing, factors and their respective Simulink subsystems.

Error-Related Factor	**Reference Value**	**Description**	**Type**	**Simulink Subsystem**
$\boldsymbol{M}_{SF}$	$1\ [-]$	Scale Factor for accelerometer ($\boldsymbol{M}_{SF,A}$) and gyroscope ($M_{SF,G,yy}$)	Deterministic Multiplicative	Sensor
M_{TS}	$0.02\ \left[\frac{\%}{°C}\right]$	Temperature Sensitivity increases with temperatures deviating from 25 °C	Deterministic Multiplicative	Sensor
$\boldsymbol{M}_{MA}$	$0\ [\%]$	Misalignment occurs due to installation errors or fabrication defects	Deterministic Multiplicative	Sensor
$\omega_{AS,G,y}$	$0.1\ \left[\frac{°/s}{g}\right]$	gyroscope acceleration sensitivity along the y-axis	Deterministic Additive	Sensor
$\omega_{RB,G,y}$	$-0.035\ \left[\frac{°}{s}\right]$	Residual bias after compensation	Deterministic Additive	Sensor
$\boldsymbol{a}_{RB,A}$	$0\ \left[\frac{m}{s^2}\right]$	Residual bias after compensation	Deterministic Additive	Sensor
$\sigma_{G,y}$	$0.128\ \left[\frac{°}{s}\right]$	The STD characterises the white gaussian gyroscope y-axes noise	Stochastic Additive	Sensor
$\sigma_{A,z}$	$0.034\ \left[\frac{m}{s^2}\right]$	The STD characterises the white gaussian accelerometer z-axes noise	Stochastic Additive	Sensor
$\sigma_{A,x}$	$0.012\ \left[\frac{m}{s^2}\right]$	The STD characterises the white gaussian accelerometer x-axes noise	Stochastic Additive	Sensor

To determine which factors of the three categories shall be analysed, a preliminary test series was conducted.

6.2 Preliminary Tests

After the 1-DoF Simulink model had been completed in its entirety and all reference values had been fixed, a preliminary test series of explorative nature, also referred to as pro forma analysis in [64], was conducted. It was used for the tuning process for the different filters (Section 5.4), to decide on the response variables, key factors and their

practicable ranges. And finally, it led to the design of the main test series. Therefore, this section will cover steps AS3 to AS6 introduced in Table 6-1.

6.2.1 Response Variables

The main response variable was chosen to be the root-mean-square-error ($RMSE$) of the true trajectory $\theta(t)$ to the desired / commanded trajectory $\theta_{com}(t)$. The $RMSE$ can be used to evaluate the performance of the whole control system by calculating the quadratic difference between $\theta(t)$ and $\theta_{com}(t)$.

$$RMSE = \sqrt{E[(\boldsymbol{\theta}_{com} - \boldsymbol{\theta})^2]} = \sqrt{\frac{1}{N_\theta}\sum_{i=1}^{N_\theta}\left(\theta_{i,com} - \theta_i\right)^2} \qquad \textbf{Eq. (6–2)}$$

Note that Eq. (6–2) and Eq. (6–3) are simplified for the 1-DoF case and therefore neglect the Euclidean norm. The $RMSE$, as well as any regression model introduced in later sections, is always indicated in °. Furthermore, N_θ represents the number of samples for the pseudo-continuous trajectories and depends on the used Simulink solver (see appendix Section C). For the chosen trajectory, established in 4.2, the $RMSE$ shall be smaller than 2.5 °, according to F35 in Table 3-3.

In [65] the mean-absolute-error (MAE, see Eq. (6–3)) and the geometric average error (GAE) are introduced as means to evaluate the error between the two trajectories. The smaller the respective value, the better the performance. This holds true for all mentioned metrics. However, the $RSME$ will always result in the largest value of the three [20].

$$MAE = E[|(\boldsymbol{\theta}_{com} - \boldsymbol{\theta})|] = \frac{1}{N_\theta}\sum_{i=1}^{N_\theta}\left|\left(\theta_{i,com} - \theta_i\right)\right| \qquad \textbf{Eq. (6–3)}$$

Since the $RMSE$ is closely related to the standard deviation and therefore considers the quadratic error, the metric possesses a higher sensitivity towards higher residuals. In other words, undesirable behavior such as, high overshoots, high estimator rise times, slow convergence of the true trajectory towards the commanded trajectory and extremely volatile behavior around the commanded trajectory will be emphasized more in the $RMSE$ than in the MAE or GAE. Hence, it can be argued, that the $RMSE$ is in favor of more aggressive trajectories, which converge quickly. However, since this characteristic is pertinent to this work, the $RSME$ was chosen to evaluate the performance of the entire control system. The $RMSE$ and all $RMSE$-related results will be indicated in °.

In [12], the $RMSE_{\hat{\theta}}$ is calculated between the estimated trajectory θ_{est} and the discretized true trajectory $\theta_{discrete}$. This allows one to isolate the filter performance from the controller performance. The sample time of the two trajectories is equivalent to the sensor sample time Δt. Therefore, the number of samples $N_{\hat{\theta}}$ is significantly smaller then the number of samples N_θ.

$$RMSE_{\hat{\theta}} = \sqrt{E\left[\left(\theta_{discrete} - \hat{\theta}\right)^2\right]} = \sqrt{\frac{1}{N_{\hat{\theta}}}\sum_{i=1}^{N_{\hat{\theta}}}\left(\theta_{i,discrete} - \hat{\theta}_i\right)^2} \qquad \textbf{Eq. (6–4)}$$

However, worse filter performance can arguably lead to worse overall performance. Therefore, the $RMSE_{\hat{\theta}}$ and the $RMSE$ are coupled, which places the $RMSE_{\hat{\theta}}$ as a secondary response variable. Secondary response variables will only be mentioned if unusual behavior is detected.

Another secondary response variable would be the overshoot, θ_{OS}, which according to F34 in Table 3-3, shall be kept under $60\,\%$ of the initial disturbance. However, the overshoot is also coupled with the $RMSE$ and therefore not qualified as a primary response variable. If the overshoot limit is surpassed in the test series but the F35 is still fulfilled, the resulting performance will still be rated.

6.2.2 Preliminary Test Results

After clarifying the rationale behind the $RMSE$ as the primary response variable, the key findings of the preliminary test series will be presented in this section. These findings were previously mentioned throughout Chapter 4 and 5.

Table 6-5: Preliminary test series findings and the resulting implications for the main test series or the subsystem design.

ID	Description	Implication
PT01	Saving gas by varying u_{bb0} without impairing the performance (see Fig. 6-1)	$u_{bb0} = 0.1$ as reference value
PT02	Mass loss due to gas consumption $m_{gc} \approx 0.178\,\mathrm{kg}$ has negligible impact on the $RMSE$	$m(t) = constant = m_0$
PT03	Optimal performance for AKF without P2PMAF	Exclude P2PMAF as a prefilter for the AKF
PT04	Performance of KF+P2PMAF comparable to CF+P2PMAF	Only evaluate CF+P2PMAF and AKF
PT05	Calculating Q, R, P_0 configuration with respective STDs satisfy performance requirements but leaves room for improvement	Use formulas from Eq. (5–29) for initial KF tuning and optimize by reducing Q with tuning coefficient C
PT06	Requirement F35-fulfilling K_C and N_{P2P} for reference values was determined	Set $K_C = 0.9987$ and $N_{P2P} = 340$ as reference values for the CF
PT07	Acceleration Sensitivity negligible for test stand applications	$\omega_{AS,G,y}$ will not be further analysed in the main test series (nuisance factor)
PT08	Significant loss in performance for varying $t_{PIDdelay}$.	$t_{PIDdelay}$ will be a further analyzed in the main test series (key factor)
PT09	F21 T_E-variation with negligible impact on $RMSE$ see Fig. E-20 and Fig. E-21.	T_E will not be analyzed in the main test series (nuisance factor)
PT10	Quantization error for the chosen fixed measurement range is negligible (E.2)	Measurement range will not be varied (nuisance factor)
PT11	Varying the residual bias was found to have negligible impact on both filters (appendix Fig. E-32 - Fig. E-35).	$\dot{\theta}_{RB,G,y}$ will not be analysed in the main test series (nuisance factor)

In PT1 it is described, that a significant amount of gas can be saved by increasing the u_{bb0} value of the BB controller. This behaviour is depicted in Fig. 6-1 with the performance depicted in Fig. 6-2. A value of $u_{bb0} = 0.1$ was chosen as the reference value, since the performance losses become noticeable after this value. If, in later design iterations, gas saving is prioritized over performance, the threshold u_{bb0} of the BB qualifies as a viable option for that.

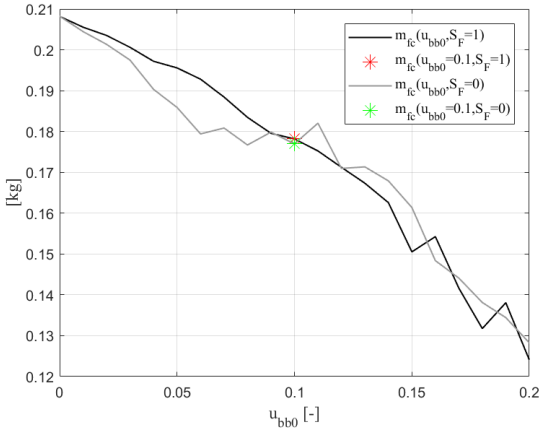


Fig. 6-1: Decrease in m_{fc} for increasing u_{bb0}. CF in grey and AKF in black.

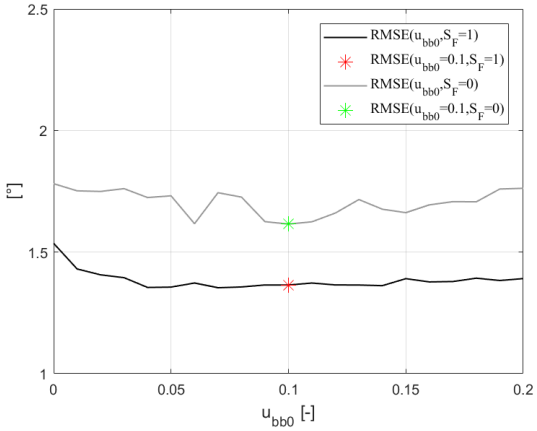


Fig. 6-2: $RMSE$ for increasing u_{bb0}. CF in grey and AKF in black.

With this, the reference values are determined. The resulting performance for the two filter configurations is shown below.

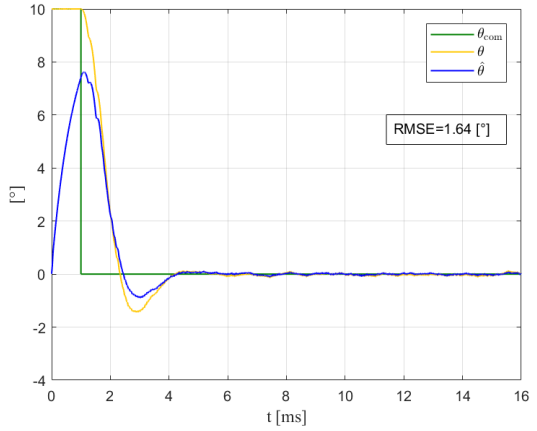


Fig. 6-3: P2PMAF+CF performance result for reference values.

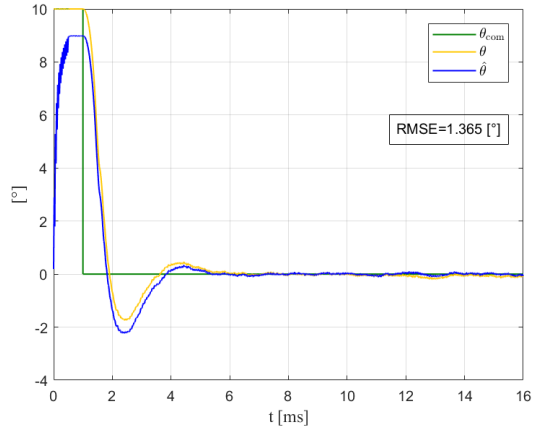


Fig. 6-4: AKF performance result for reference values.

Fig. 6-3 and Fig. 6-4 show that, both filter configurations satisfy F35 under reference condtitions. Thus, the reference performance is $RMSE_{ref} = 1.640\,°$ for the P2PMAF+CF configuration and $RMSE_{ref} = 1.365\,°$ for the AKF configuration. Results for the other two trajectories established in Section 4.2 can be found in the appendix Section E.1. The reliability of this performance results shall be tested by varrying the key factors which will be established in the next section.

6.2.3 Key Factors, Levels and Ranges

According to [60], the selected key factors should be varied over a practicable range on an appropriate amount of levels. Levels can be understood as the resolution of the input data. For two levels, the range of a factor would be divided into a high and a low value, which is computationally efficient; however, information would be lost. The input for the experiment will be defined in the next sections.

6.2.3.1 Key Factor Selection

The established factor categories allow for the following distinction when deciding on the key factors:

Most internal factors are customizable, with restrictions only applying to f_{sensor}, f_{bb}, $\boldsymbol{\omega}_{range}$ and $\boldsymbol{a}_{range}$, and therefore are able to react to a certain circumstance. As established in Section 5.4, factors can be adjusted to keep the performance within the requirements boundary. Therefore, they will be used to formulate the tuning rules, but will not be labelled as key factors.

For the external factors, the design and implementation team suggested possible factors of interest. These include the actuator switching time, t_{st}, the delay time after the PID element, $t_{PIDdelay}$, the initial mass, m_0, the maximum actuator force, F_{max}, and a change of the actuator lever arm, l_A. However, l_A and F_{max} are inversely proportional to m_0,

$$m_0 \sim \frac{1}{l_A F_{max}}. \qquad \textbf{Eq. (6–5)}$$

This can be extracted from Eq. (4–1). Therefore, only m_0 will be considered for the evaluation. Additionally, the maximum controllable deflection angle will be evaluated by varying θ_{ES}. The temperature sensibility was a suggested factor as well, but was ultimately ruled out during the preliminary tests (see Fig. E-20 and Fig. E-21).

For the error-related factors, the scale factor, $\boldsymbol{M}_{SF}$, and the random noise process, σ, will be analysed. The effect of misalignment and all other multiplicative sensor errors are directly proportional to the scale factor, and therefore do not provide any additional evaluation information. The listed error-related factors were chosen since they are expected to have the highest influence on the performance and cannot be compensated by calibration without profound testing and sophisticated sensor error models. Further information on compensating the scale factor of an IMU by calibration is given in [66]

This selection of six factors marks the first iteration of key factors, which will be further refined, after the main test series, to answer research question Q2.

6.2.3.2 Input Factor Levels

The number of levels, L, in an experiment can be understood as the resolution of the acquired data. To obtain higher order relations between input and output, there must be at least three levels to an experiment. Increasing the number of levels, however, will evidently vastly increase the computational effort Eq. (6–1). The behavior of the model to the proposed input factors is completely unknown and therefore, the input shall be randomized across enough levels and a wide enough range. To model this

uncertainty, an arbitrary input modelled by its PDF with a standard deviation is chosen. To capture the effect of inputs 3σ away from the mean, on the response, the sample size of the Monte Carlo simulation must be $L \geq 371$. This is defined by the characteristic of the chosen input distribution for which $99.73\,\%$ of samples must lie within the 3σ range (see appendix Chapter B). Therefore, the sample size for the main experiment was chosen to be $L = 400$, to balance informativeness with available time and resources.

Preliminary tests were conducted with a fourth of the main test sample size $L_{pre} = 100$, which is appropriate according to [64].

6.2.3.3 Input Factor Ranges

For this evaluation, the range covers the factor space of interest in which either $68.27\,\%$ or $100\,\%$ of the input samples are located, depending on the input function used for the sensitivity analysis. Section 6.3 will elaborate on the two different distribution used for the evaluation. The range for the respective key factors was either taken from the datasheets [14, 23] or estimated with calculations by the design team [26, 32]. As an example:

$$\sigma_x = x_{ref} R_x \qquad \textbf{Eq. (6–6)}$$

will define one STD of the gaussian normal distributed input

$$\delta x = \mathcal{N}\left[x_{ref}, \sigma_x\right]. \qquad \textbf{Eq. (6–7)}$$

One standard deviation can, therefore, also be referred to as the $68.27\,[\%]$ -error, since this exact percentage of samples of the random input lie within $\pm\sigma_x$. The probability density function (PDF) is given in the appendix Section B.2.

6.2.4 Experiment Design Choice

To characterize the performance of the two filter configurations in the simulation a one-factor-at-a-time (OFAT) analysis was conducted. OFAT is a commonly used method to analyze sensitivity due to its simplicity and practicability. Here, one factor is varied while all other factors of an experiment are kept at a constant, reference value [60]. In this work, information about the influence of one factor on the performance of the respective filter can be obtained. After analyzing the factor of interest, it is returned to the reference value and the next input factor can be varied, while keeping all other values constant.

OFAT experiments depend upon guesswork, luck, experience and intuition for its success. This may often lead to unreliable, inefficient and time-consuming experiments and may yield false optimum conditions for the process [60]. Therefore, OFAT is strictly not considered a DOE method and often out-performed by such methods in terms of efficiency, information yield and exploring the entire factor space [67]. Therefore, in an OFAT experiment, interactions between input factors might remain undetected. In DOE methods, such as fractional factorial method Box-Behnken from [62], input factors resolution is simplified to three levels, a high, a low and a central point level. This allows the method to explore a vast multi-factorial design space to obtain an optimal configuration with relatively few runs. However, the goal of this experiment is not to find the optimal configuration, but to obtain the performance sensitivity of the system

regarding key factors. The low resolution of fractional factorial methods may impair the significance of the statement regarding the sensitivity of the system towards one factor. In the context of this simulation experiment, the simplicity of OFAT allows, to give a randomized, high sample size, statistical input, in the form of a any distribution, described by the previously determined range. The response can be analyzed by using statistics and regression models.

In summary, the two filter configurations will be characterized by analyzing the sensitivity of the $RMSE$ towards one arbitrary key factor input at a time.

6.3 Sensitivity Analysis

After choosing a design for the simulation experiment, the main test series was conducted. To parse the effect of the chosen key factors on the performance of the control system a sensitivity analysis is performed for statistical evaluation.

In the context of numerical evaluation of mathematical models, sensitivity can be described as the relation between a change in the input to the resulting change in the output. In practical modelling, the sensitivity analysis is carried out by changing the input parameters. The corresponding response on the selected input δx is observed. Thus, the sensitivity, S_x, towards an input, x, is defined by the partial derivative with respect to a reference value [68]:

$$S_x = \frac{x_{ref}}{y_{ref}} \frac{\partial y}{\partial x}\Big|_{x_{ref}} = \frac{x_{ref}}{RMSE_{ref}} \frac{\partial RMSE(x)}{\partial x}\Big|_{x_{ref}}. \qquad \textbf{Eq. (6–8)}$$

This sensitivity computation will be labelled partial derivative sensitivity (PDS), where the quotient, $x_{ref}/RMSE_{ref}$, is introduced to normalize the coefficient by removing the units [69]. Since the derivative is computed at a fixed reference point, it is considered a local method and therefore not recommended to explore the whole input space of possibly nonlinear systems [61]. Additionally, it does not consider the different ranges in the input. However, it can be modified to analyse the effect of several arbitrary inputs on an arbitrary output. For this purpose, the partial derivative at the reference point is normalized by the coefficient σ_x/σ_y, which is referred to as the square root of the importance index in [69]. Whereby, σ_x represents the STD of one input and σ_y represents the STD of the response distribution for varying all factors at once. This metric is referred to as the sigma-normalized derivatives sensitivity (SNS), S_x^σ, and is often used for the statistical analysis of Monte Carlo simulations. The squared can be formulated with the equation [61]:

$$(S_x^\sigma)^2 = \left(\frac{\sigma_x}{\sigma_y} \frac{\partial y}{\partial x}\Big|_{x_{ref}}\right)^2 = \left(\frac{\sigma_x}{\sigma_{RMSEtot}} \frac{\partial RMSE(x)}{\partial x}\Big|_{x_{ref}}\right)^2. \qquad \textbf{Eq. (6–9)}$$

The higher the squared SNS is, the higher is the share of the respective input in the output distribution. If the response function is a linear combination of the inputs, the sum of the squared SNSs will equate to one. If this is not the case and non-additive or nonlinear terms exist in the response equation, the sum of the squared SNSs will be less than one [70].

For this work, the squared SNS shall allow to rank the chosen input factors, relative to each other, for one filter, regardless of the input distribution and range. Additionally,

the statistics of the response for varying one factor at a time will give absolute values like the standard deviation or the mean.

Not all key factor inputs can be varied randomly without restriction. The computational time delay, the end stop angle as well as the STD of the sensor noise shall be varied over a vast range to identify the limits of the systems. However, they cannot be negative and therefore a normally distributed input is not appropriate. It was chosen to model the input with an equal / uniform distribution. It will be further described in 6.3.2 and in the appendix Section B.3.

Both input distributions will be analyzed with the mentioned sensitivity metrics. The partial derivative will be calculated by using a linear regression model fitted to the response scatter plot, resulting from a single input variation. If a linear model is not an appropriate fit, the partial derivative is calculated at x_{ref}. Since the equally distributed values will cover a vast design space, additionally, the maximum allowed values, for the respective input factor, to still fulfill F35 will be provided. These results are presented throughout Sections 0 to 6.3.2.3, while all results together with the SNS values for all factors and filters will be summarized in 6.3.3. After introducing yet another level of distinction for evaluating the key factors, an overview of the whole factor classification process is given in the figure below.

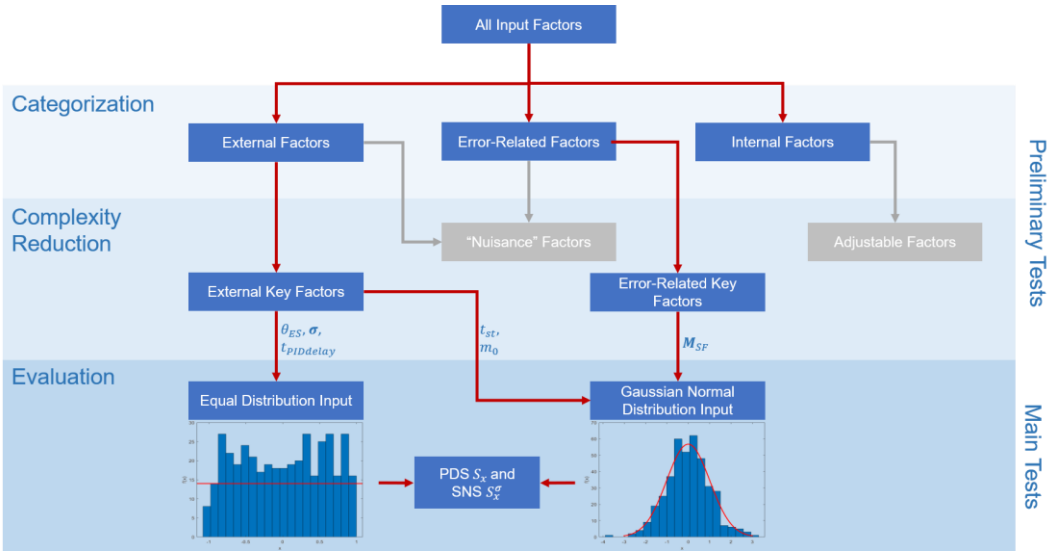


Fig. 6-5: Overview of classification and evaluation process of the input factors.

Further, both filter configurations will use the same arbitrary generated input, for varying one factor at a time, as well as for varying all factors at once, to allow for comparable results.

6.3.1 Normal Distribution Input

The normal distributed arbitrary input is generated with the MATLAB function `delta_x=normrnd(x_ref,x_ref*R,L,1)`. The PDF is given in the appendix Eq. (B–5). The output will be analyzed by using the characteristic parameters μ_{RMSE} and σ_{RMSE}, resulting from the normal distribution fit function `mu_sigma=fitdist(x,y,'normal')`, and by fitting an appropriate regression function to the scatter plot. The acceptable limit to the $RMSE$, demanded by F35, will be visible in the scatter plot.

6.3.1.1 Actuator Switching Time

For the actuator switching time, t_{st}, the input distribution is calculated with a range of $R_{t_{st}} = 30\,\%$. This value was assumed to cover the tolerance specified in [23]. The MATLAB function with the corresponding values is:

```
delta_t_st = normrnd(0.0035,0.0035*0.3,400,1);
```

Scatter plot and histogram with PDF for the CF+P2PMAF is shown below.

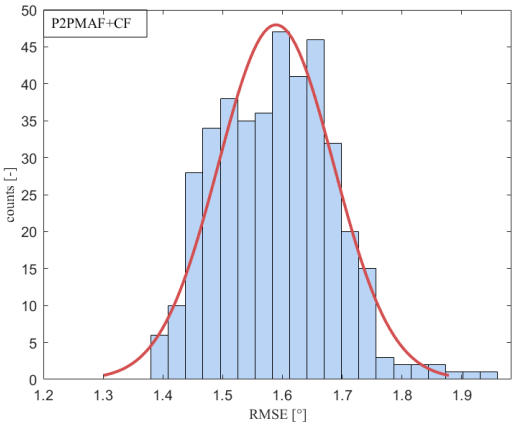


Fig. 6-6: Histogram for t_{st}, $\mu_{RMSE} = 1.589\,°$ and $\sigma_{RMSE} = 0.096\,°$.

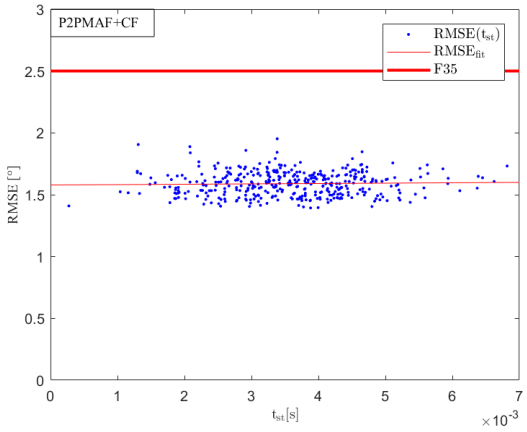


Fig. 6-7: $RMSE$ scatter plot for t_{st}, and linear regression fit.

Fig. 6-6 indicates that the performance of the CF configuration has a probability of 68.27 % to fall within one STD, $\sigma_{RMSE} = 0.096\,°$, from the mean, for the given $R_{t_{st}}$.

$$p(RMSE = 1.589\,[°] \pm 0.096\,[°]) = 68.27\,[\%]. \qquad \textbf{Eq. (6–10)}$$

The linear regression of the scatter plot reveals the function:

$$RMSE_{fit}(t_{st}) = 3.003 t_{st} + 1.579. \qquad \textbf{Eq. (6–11)}$$

During the simulation, it was observed, that occasionally the zero-crossing detection was deactivated. This can be observed in the $RMSE_{\hat{\theta}}$ scatter plot in the appendix Fig. E-22. This simulation issue may lead to a higher variance in the output. Next, the histogram with the PDF and the scatter plot for the AKF is shown below.

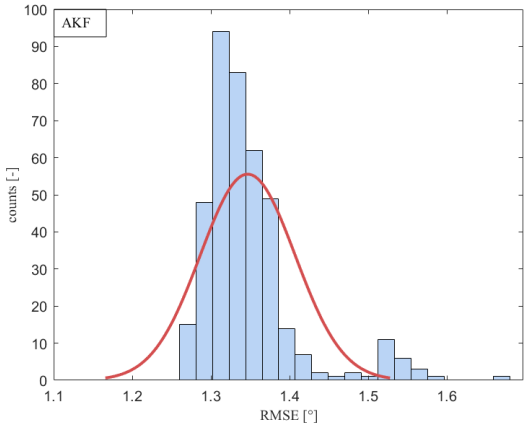


Fig. 6-8: $RMSE$ histogram for t_{st}, $\mu_{RMSE} = 1.347\,°$ and $\sigma_{RMSE} = 0.060\,°$.

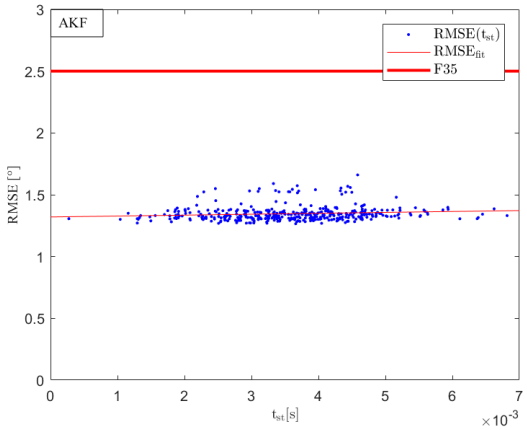


Fig. 6-9: $RMSE$ scatter plot for t_{st} and linear regression fit.

Fig. 6-8 indicates that the performance of the AKF configuration has a probability of 68.27 % to fall within one standard deviation $\sigma_{RMSE} = 0.060\,°$ from the mean, for $R_{t_{st}}$.

$$p(RMSE = 1.347\,[°] \pm 0.060\,[°]) = 68.27\,[\%]. \qquad \textbf{Eq. (6–12)}$$

The linear regression of the scatter plot reveals the function:

$$RMSE_{fit}(t_{st}) = 7.390 t_{st} + 1.320. \qquad \textbf{Eq. (6–13)}$$

The output results for the two filter configurations are summarized in the table below.

Table 6-6: Output statistics for both filter configurations with key factor t_{st}.

Filter	Range	STD	Mean	PDS
P2PMAF+CF	30 [%]	0.096 [°]	1.589 [°]	0.006 [–]
AKF		0.060 [°]	1.347 [°]	0.018 [–]

6.3.1.2 Weight

For the starting mass, m_0, the input distribution is calculated with a range of $R_{m_0} = 0.356\,\%$. This value is calculated by dividing the gas consumption, calculated with Eq. (4–19), by the starting mass. It resembles the expected change in total mass during a test run. The MATLAB function with the corresponding values is:

```
delta_m_0 = normrnd(5,5*0.0356,400,1);
```

The histogram with the pdf for the CF+P2PMAF configuration is shown in Fig. 6-10 and the scatter plot in Fig. 6-11.

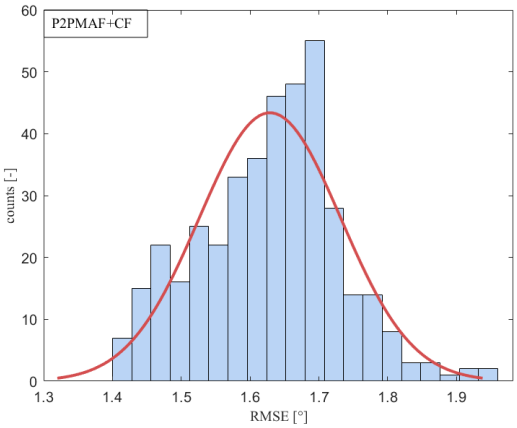


Fig. 6-10: $RMSE$ histogram for m_0, $R = 3.56\,\%$, $\mu_{RMSE} = 1.629\ °$ and $\sigma_{RMSE} = 0.103\ °$.

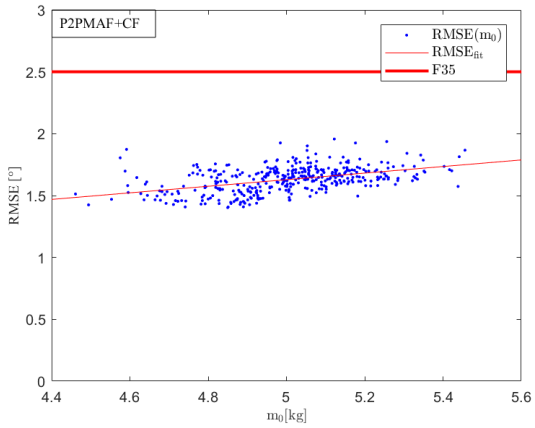


Fig. 6-11: $RMSE$ scatter plot for m_0, $R = 3.56\,\%$ and a linear fit.

The histogram indicates that the performance of the CF configuration has a probability of $68.27\,\%$ to fall within one standard deviation $\sigma_{RMSE} = 0.103\ °$ from the mean, for R_{m_0}.

$$p(RMSE = 1.629\ [°] \pm 0.103\ [°]) = 68.27\ [\%]. \qquad \textbf{Eq. (6–14)}$$

A linear model was fitted to the scatter plot. It can be expressed with the function:

$$RMSE_{fit}(m_0) = 0.266 m_0 + 0.299. \qquad \textbf{Eq. (6–15)}$$

The histogram with the PDF for the AKF configuration is shown in Fig. E-26 and the scatter plot in Fig. E-27.

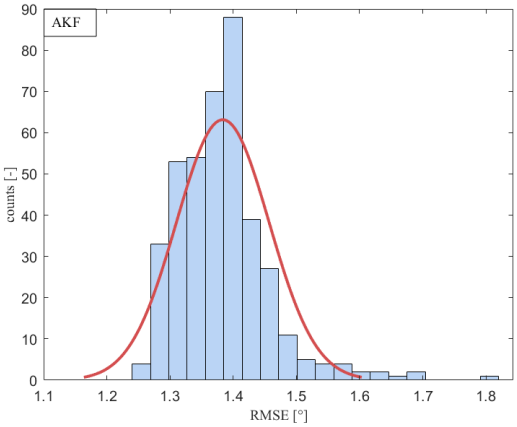


Fig. 6-12: $RMSE$ histogram for m_0, $R = 3.56\,\%$, $\mu_{RMSE} = 1.383\ °$ and $\sigma_{RMSE} = 0.073\ °$.

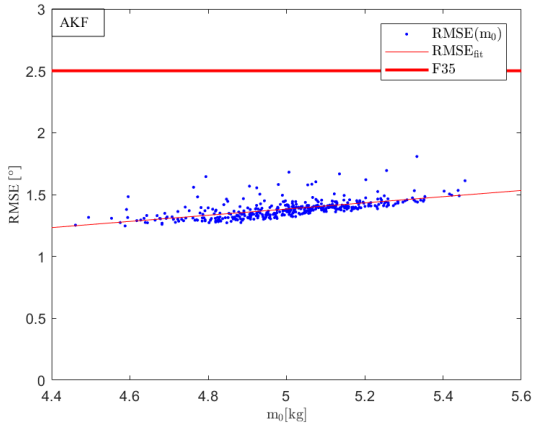


Fig. 6-13: $RMSE$ scatter plot for m_0, $R = 3.56\,\%$ and a linear fit.

The histogram indicates that the performance of the AKF configuration has a probability of $68.27\,\%$ to fall within one standard deviation $\sigma_{RMSE} = 0.073\ °$ from the mean, for the given input range.

$$p(RMSE = 1.383\ [°] \pm 0.073\ [°]) = 68.27\ [\%]. \qquad \textbf{Eq. (6–16)}$$

A linear model was fitted to the scatter plot. It can be expressed with the function:

$$RMSE_{fit}(m_0) = 0.250m_0 + 0.133. \qquad \textbf{Eq. (6–17)}$$

If the range is extended to $R_{m_0} = 5\ \%$, the regression function changes from a linear model to a second-order exponential model (see Eq. (E–21) and Eq. (E–24)). When solving the equation for the value $RMSE = 2.5\ °$, the maximum weight the RCS can control without adjusting the PID values can be obtained. For the P2PMAF+CF configuration the maximum weight is $m_{0,max,CF} = 5.963\ kg$ (see Eq. (E–22)) while for the AKF, it is $m_{0,max,AKF} = 6.185\ \mathrm{kg}$ (see Eq. (E–25)). Further information regarding these test results can be found in the appendix Section E.5. Calculating the PDS at the reference point gives results comparable to the linear model results for the lower range. Thus, reinforcing the findings. The results for both ranges and filters are comprised in the table below.

Table 6-7: Output statistics for both ranges and filter configurations with key factor m_0.

Filter	Range	STD	Mean	PDS
P2PMAF+CF	3.56 [%]	0.103 [°]	1.629 [°]	0.810 [–]
AKF		0.073 [°]	1.383 [°]	0.916 [–]
P2PMAF+CF	5 [%]	0.117 [°]	1.627 [°]	0.707 [–]
AKF		0.104 [°]	1.393 [°]	0.854 [–]

It was observed that varying the centre of mass, therefore changing l_{CM}, generates sensitivity results comparable to the δm_0 results. The δl_{CM} results are provided in the appendix Section E.6.

6.3.1.3 Scaling

For the scale factor, $\boldsymbol{M}_{SF}$, the input distribution is calculated with a range of $R_{\boldsymbol{M}_{SF}} = 3\ \%$. This value was assumed to cover the tolerance specified in [14]. The scale factor of the gyroscope is examined independently of the accelerometer scale factor. The MATLAB expressions with the corresponding values are:

```
delta_M_SF_G_yy = normrnd(1,1*0.03,400,1);
delta_M_SF_A = normrnd(1,1*0.03,400,2);
```

The gyroscope scale factor is examined first. The histogram with the PDF for the CF+P2PMAF configuration is shown in Fig. 6-6 and the scatter plot in Fig. 6-7.

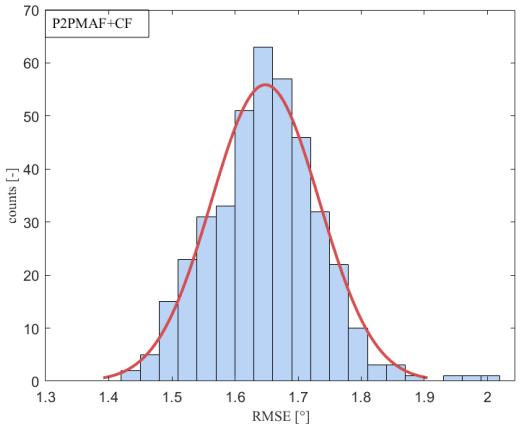


Fig. 6-14: $RMSE$ histogram for $M_{SF,G,yy}$, $\mu_{RMSE} = 1.648\,°$ and $\sigma_{RMSE} = 0.086\,°$.

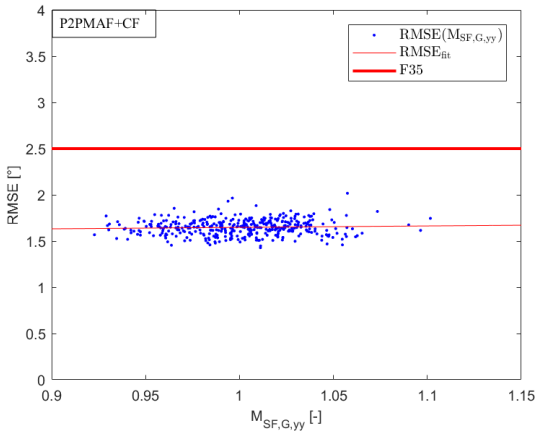


Fig. 6-15: $RMSE$ scatter plot for $M_{SF,G,yy}$ and linear regression fit.

Fig. 6-6 indicates that the performance of the CF configuration has a probability of 68.27 % to fall within one standard deviation, $\sigma_{RMSE} = 0.086\,°$, from the mean, for $R_{\boldsymbol{M_{SF}}}$.

$$p(RMSE = 1.648\,[°] \pm 0.086\,[°]) = 68.27\,[\%]. \qquad \textbf{Eq. (6–18)}$$

The linear regression of the scatter plot reveals the function:

$$RMSE_{fit}\left(M_{SF,G,yy}\right) = 0.154 M_{SF,G,yy} + 1.494. \qquad \textbf{Eq. (6–19)}$$

The histogram with the pdf for the AKF configuration is shown in Fig. 6-8 and the scatter plot in Fig. 6-9.

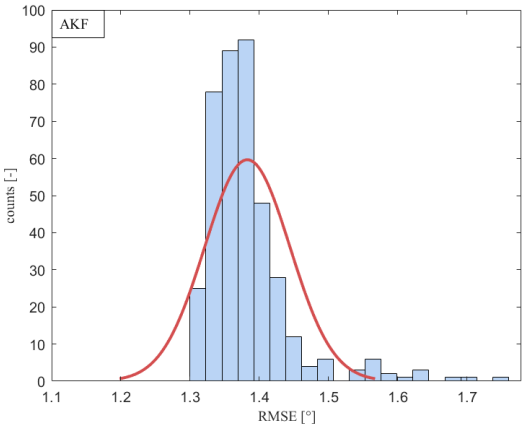


Fig. 6-16: $RMSE$ histogram for $M_{SF,G,yy}$, $\mu_{RMSE} = 1.383\,°$ and $\sigma_{RMSE} = 0.062\,°$.

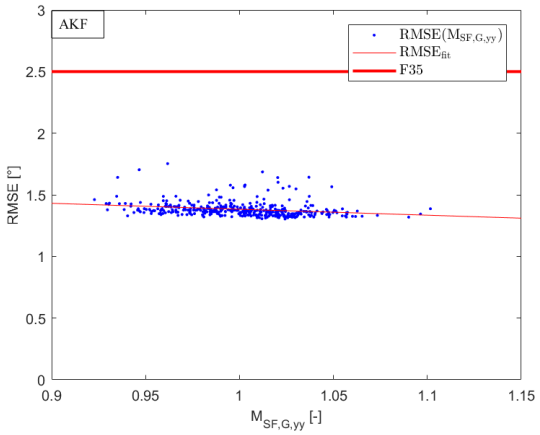


Fig. 6-17: $RMSE$ scatter plot for $M_{SF,G,yy}$ and linear regression fit.

The histogram indicates that the performance of the AKF configuration has a probability of 68.27 % to fall within one standard deviation, $\sigma_{RMSE} = 0.062\,°$, from the mean, for $R_{\boldsymbol{M_{SF}}}$.

$$p(RMSE = 1.383\,[°] \pm 0.062\,[°]) = 68.27\,[\%]. \qquad \textbf{Eq. (6–20)}$$

The linear regression of the scatter plot reveals the function:

$$RMSE_{fit}\left(M_{SF,G,yy}\right) = -0.483 M_{SF,G,yy} + 1.866. \qquad \textbf{Eq. (6–21)}$$

For the accelerometer, two axes are of interest. The x- and the z- axes of the actuator frame (A-frame). First the P2PMAF+CF configuration is examined. The histogram and the 3D scatter plot with a multivariate linear regression fit is shown below.

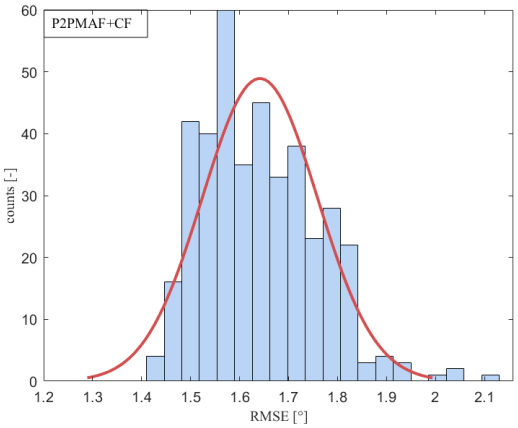


Fig. 6-18: $RMSE$ histogram for $\boldsymbol{M}_{SF,A}$, $\mu_{RMSE} = 1.641\,°$ and $\sigma_{RMSE} = 0.117\,°$.

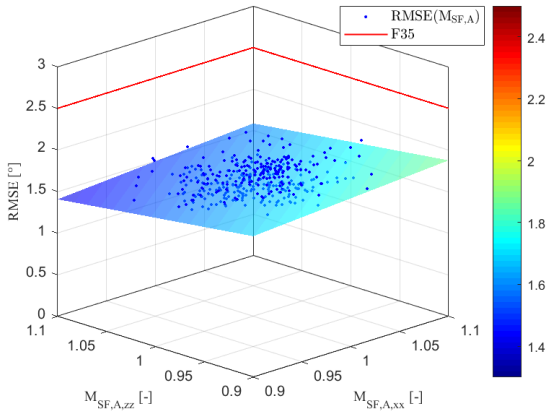


Fig. 6-19: $RMSE$ scatter plot for $\boldsymbol{M}_{SF,A}$ and multivariate linear regression fit.

Fig. 6-18 indicates that the performance of the AKF configuration has a probability of 68.27 % to fall within one standard deviation $\sigma_{RMSE} = 0.117\,°$ from the mean, for $R_{\boldsymbol{M}_{SF}}$.

$$p(RMSE = 1.641\,[°] \pm 0.117\,[°]) = 68.27\,[\%]. \qquad \textbf{Eq. (6–22)}$$

The multivariate linear regression fit of the 3D scatter plot reveals the function:

$$RMSE_{fit}(\boldsymbol{M}_{SF,A}) = 0.907 M_{SF,A,xx} - 1.433 M_{SF,A,zz} + 2.167. \qquad \textbf{Eq. (6–23)}$$

Next, the AKF configuration is examined. The histogram is shown in Fig. 6-20 and the 3D scatter plot with a multivariate linear regression fit is displayed in Fig. 6-21.

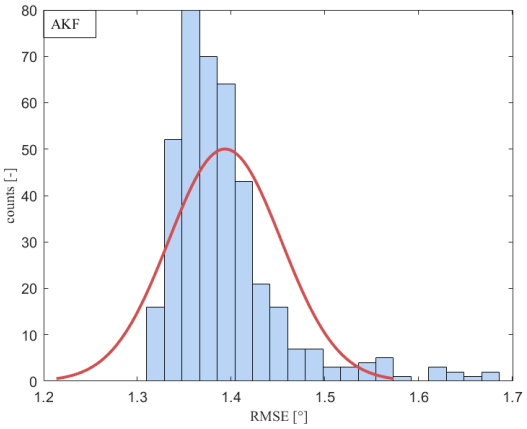


Fig. 6-20: $RMSE$ histogram for $\boldsymbol{M}_{SF,A}$, $\mu_{RMSE} = 1.393\,°$ and $\sigma_{RMSE} = 0.060\,°$.

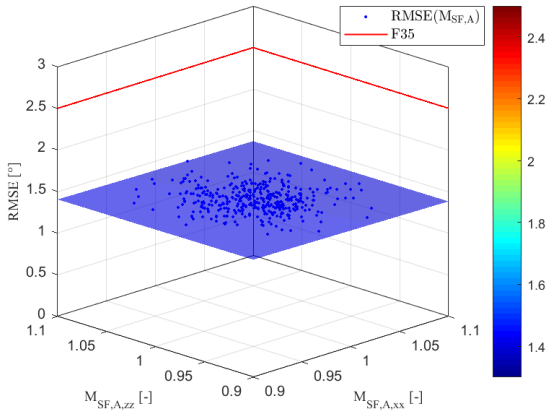


Fig. 6-21: $RMSE$ scatter plot for $\boldsymbol{M}_{SF,A}$ and multivariate linear regression fit.

Fig. 6-20 indicates that the performance of the AKF configuration has a probability of 68.27 % to fall within one standard deviation, $\sigma_{RMSE} = 0.060\,°$, from the mean, for $R_{\boldsymbol{M}_{SF}}$.

$$p(RMSE = 1.393\,[°]\ \pm 0.060\,[°]) = 68.27\,[\%]. \qquad \textbf{Eq. (6–24)}$$

The linear regression of the scatter plot reveals the function:

$$RMSE_{fit}(\boldsymbol{M}_{SF,A}) = -0.152M_{SF,A,xx} - 0.036M_{SF,A,zz} + 1.581. \quad \textbf{Eq. (6–25)}$$

The output results for the two filter configurations are summarized in the table below.

Table 6-8: Results of the two filter configurations for the key factor $\boldsymbol{M}_{SF}$.

Filter	Factor	Range	STD	Mean	PDS
P2PMAF+CF	$M_{SF,G,yy}$	3 [%]	0.086 [°]	1.648 [°]	0.094 [−]
AKF			0.062 [°]	1.383 [°]	−0.354 [−]
P2PMAF+CF	$M_{SF,A,xx}$		0.117 [°]	1.641[°]	0.553 [−]
	$M_{SF,A,zz}$				−0.874 [−]
AKF	$M_{SF,A,xx}$		0.060 [°]	1.393 [°]	−0.111 [−]
	$M_{SF,A,zz}$				−0.026 [−]

6.3.2 Equal Distribution Input

The equal distribution possesses the property that all samples lie within the limits x_{min} and x_{max}, which defines the input range. Furthermore, all inputs have the same probability of occurring. For more information refer to the appendix Section B.3. In MATLAB, such confined arbitrary input distribution can be generated with the expression:

```
delta_x = x_min + (x_max-x_min) * rand(L,1);
```

It will be analyzed by the SNS in Section 6.3.3 and by fitting an appropriate regression function to the resulting $RMSE$ scatter plot. The acceptable limit to the $RMSE$, demanded by F35, will be shown in the scatter plot. Output statistics for the equal distribution are not representative and will therefore not be analyzed.

6.3.2.1 Initial Deflection

For the initial deflection of the pendulum, which is equal to the end stop angle, θ_{ES}, the maximum controllable deflection with the reference settings shall be found. Therefore, the input is varied over a range of $8° \leq R_{\theta_{ES}} \leq 12°$. The corresponding MATLAB expression is:

```
delta_theta_ES = 8 + (12-8) * rand(400,1);
```

The result is shown in Fig. 6-22 and Fig. 6-23 for the P2PMAF+CF and AKF configuration respectively.

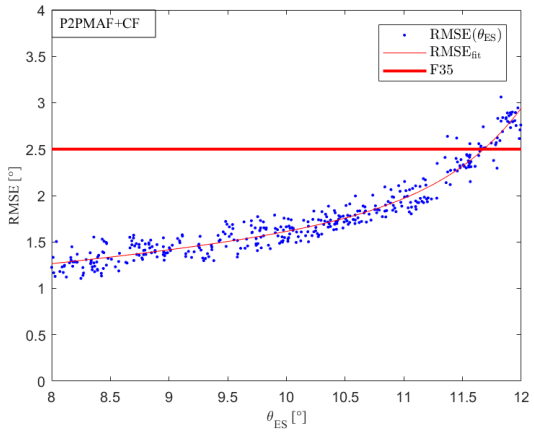


Fig. 6-22: CF $RMSE$ scatter plot for θ_{ES} a second-order exponential fit.

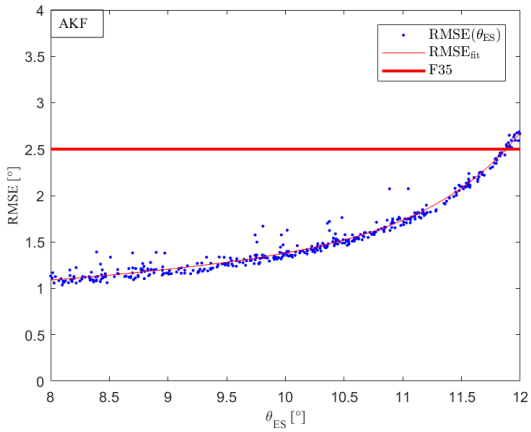


Fig. 6-23: AKF $RMSE$ scatter plot for θ_{ES} and a second-order exponential fit.

A second-order exponential model was fitted to the P2PMAF+CF scatter plot. It can be approximated with the function:

$$RMSE_{fit,CF}(\theta_{ES}) = 0.545e^{0.105\theta_{ES}} + 2.811 * 10^{-8}e^{1.451\theta_{ES}}. \quad \textbf{Eq. (6–26)}$$

For the AKF scatter plot the same model results in:

$$RMSE_{fit,AKF}(\theta_{ES}) = 0.542e^{0.086\theta_{ES}} + 4.413 * 10^{-7}e^{1.231\theta_{ES}}. \quad \textbf{Eq. (6–27)}$$

Solving the functions for the value defined by requirement F35 returns

$$\theta_{ES}\left(RMSE_{fit,CF} = F35 = 2.5\ [°]\right) = 11.680\ [°], \quad \textbf{Eq. (6–28)}$$

for the P2PMAF+CF configuration and

$$\theta_{ES}\left(RMSE_{fit,AKF} = 2.5\ [°]\right) = 11.878\ [°], \quad \textbf{Eq. (6–29)}$$

For the AKF configuration. Therefore, formulating the maximum allowed computational delay for the respective filter under reference setting.

6.3.2.2 Measurement Noise

The measurement noise is characterized by the STD of the respective sensor. The goal of this section is to quantify the performance deterioration of the control system for a STD higher than the one measured in Section 4.5. Since the KF algorithm is provided with the variance in the measurement for tuning, a superior performance of the KF compared to the CF is expected. The input is varied over a range of $0.1\ °/s \leq \delta\sigma_{G,y} \leq 0.5\ °/s$, and $0.01\ \mathrm{m/s^2} \leq \delta\boldsymbol{\sigma}_A \leq 0.1\ \mathrm{m/s^2}$, which is appropriate for consumer grade IMUs [12]. The gyroscope noise level will be examined independently from the accelerometer noise levels. The corresponding MATLAB expressions are:

```
delta_sigma_G_y = 0.1 + (0.5-0.1) * rand(400,1);
delta_sigma_A = 0.01 + (0.01-0.01) * rand(400,2);
```

The result is shown in Fig. 6-22 and Fig. 6-23 for the P2PMAF+CF and AKF configuration respectively.

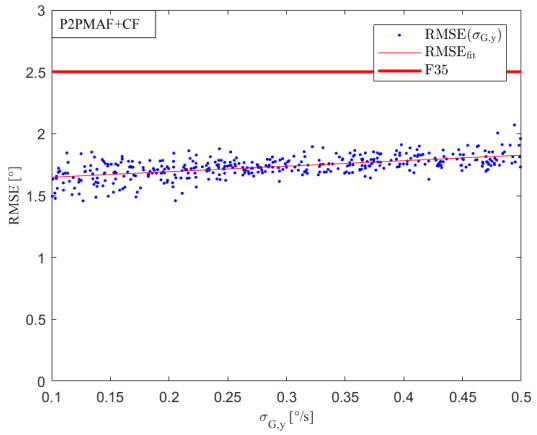


Fig. 6-24: CF $RMSE$ scatter plot for $\sigma_{G,y}$ with linear regression model.

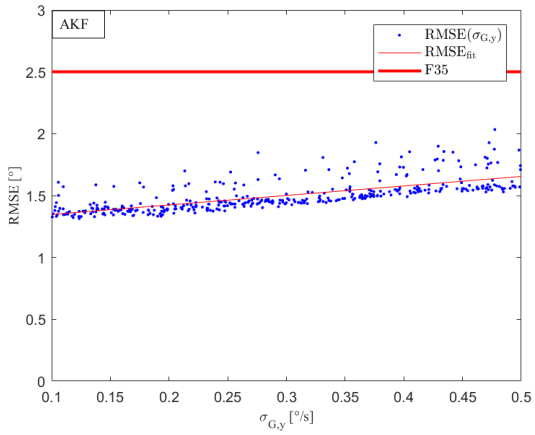


Fig. 6-25: AKF $RMSE$ scatter plot for $\sigma_{G,y}$ with linear regression model.

A linear model was fitted to the P2PMAF+CF scatter plot. It can be approximated with the function:

$$RMSE_{fit,CF}(\sigma_{G,y}) = 0.447\sigma_{G,y} + 1.603. \qquad \textbf{Eq. (6–30)}$$

For the AKF scatter plot the same model results in:

$$RMSE_{fit,AKF}(\sigma_{G,y}) = 0.767\sigma_{G,y} + 1.271. \qquad \textbf{Eq. (6–31)}$$

Solving the functions for the value defined by requirement F35 returns

$$\sigma_{G,y}(RMSE_{fit,CF} = 2.5\ [°]) = 2.008\ \left[\frac{°}{s}\right], \qquad \textbf{Eq. (6–32)}$$

for the P2PMAF+CF configuration and

$$\sigma_{G,y}(RMSE_{fit,AKF} = 2.5\ [°]) = 1.603\ \left[\frac{°}{s}\right], \qquad \textbf{Eq. (6–33)}$$

for the AKF configuration. Therefore, formulating the maximum allowed noise level for the respective filter under reference settings. These values must be treated with caution, since a linear model is assumed, which might not be the case at noise levels $\sigma_{G,y} > 0.5\ °/s$.

For the accelerometer there are two axes of interest. The x- and the z- axes of the actuator frame (A-frame). First the P2PMAF+CF configuration is examined. The graphs below show the 3D scatter plots with a multivariate linear regression fit for both filter configurations.

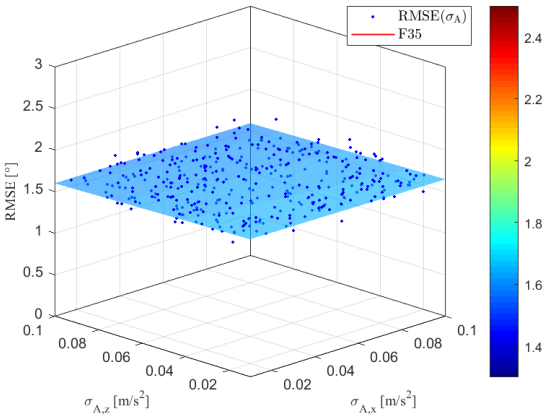


Fig. 6-26: P2PMAF+CF $RMSE$ scatter plot for $\boldsymbol{\sigma_A}$ with multivariate linear regression fit.

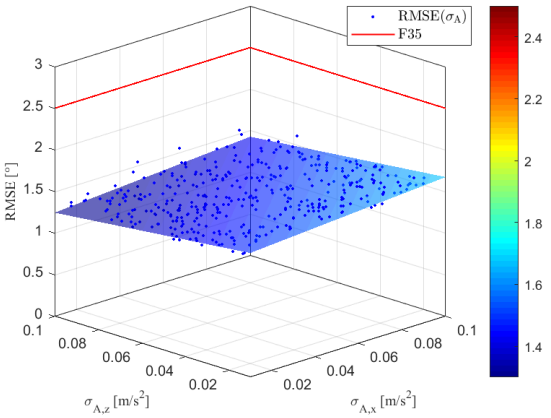


Fig. 6-27: AKF $RMSE$ scatter plot for $\boldsymbol{\sigma_A}$ with multivariate linear regression fit.

The multivariate linear regression model of the 3D scatter plot reveals the function:

$$RMSE_{fit,CF}(\boldsymbol{\sigma_A}) = -0.099\sigma_{A,x} - 0.637\sigma_{A,z} + 1.664, \qquad \textbf{Eq. (6–34)}$$

for the P2PMAF+CF configuration and

$$RMSE_{fit,AKF}(\boldsymbol{\sigma_A}) = 2.028\sigma_{A,x} - 2.726\sigma_{A,z} + 1.498, \qquad \textbf{Eq. (6–35)}$$

for the AKF configuration.

The intersection function between the surface and the $RMSE$ value defined by requirement F35 cannot be found for this variation range. In the appendix the maximum noise input was increased fivefold to obtain the intersection functions:

$$\sigma_{A,x}\left(RMSE_{fit,CF} = 2.5\ [°]\right) = 0.663 - 0.408\sigma_{A,y}, \qquad \textbf{Eq. (6–36)}$$

for the P2PMAF+CF configuration and

$$\sigma_{A,x}\left(RMSE_{fit,AKF} = 2.5\ [°]\right) = 0.548 + 0.061\sigma_{A,y}, \qquad \textbf{Eq. (6–37)}$$

for the AKF configuration. The scatter plots for the increased maximum noise input can be found in the appendix 0.

6.3.2.3 Computational Delay

The input distribution for the computational delay after the PID-element is generated with the `randi([x_min,x_max],L,1)` MATLAB function which generates an equally distributed series of integers within a certain range. This function was chosen because it was observed that, decimal numbers smaller than the sensor frequency caused problems within the simulation, resulting in the control system failing to control the pendulum. Therefore, the input expression is formulated as:

```
delta_t_PIDdelay = randi([1,120],400,1)/1000;
```

The maximum delay time $t_{PIDdelay,max} = 0.12\ s$ was chosen to be large enough to bring the control system to failure. Therefore, the input is varied over a range of $0.001\ s \leq \delta t_{PIDdelay} \leq 0.12\ s$. The result is shown in Fig. 6-28 and Fig. 6-29 for the P2PMAF+CF and AKF configuration respectively.

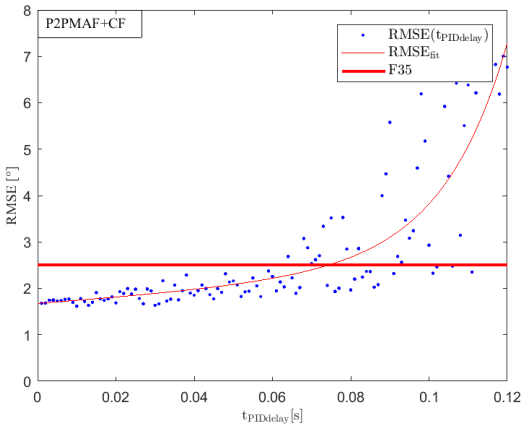


Fig. 6-28: CF $RMSE$ scatter plot for $t_{PIDdelay}$ and a second-order exponential fit.

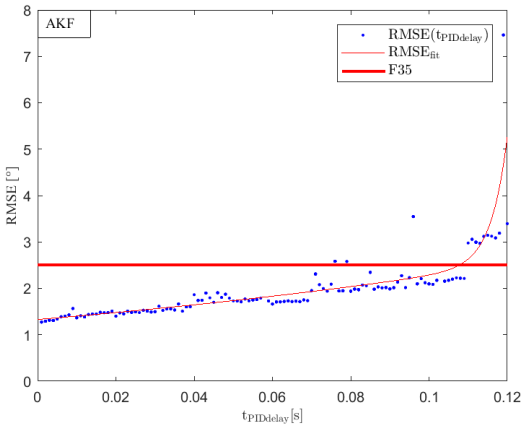


Fig. 6-29: AKF $RMSE$ scatter plot for $t_{PIDdelay}$ and a second-order exponential fit.

A second-order exponential model was fitted to the P2PMAF+CF scatter plot. It can be approximated with the function:

$$RMSE_{fit,CF}(t_{PIDdelay}) = 1.672e^{3.759t_{PIDdelay}} + 0.003e^{60.669t_{PIDdelay}}. \quad \textbf{Eq. (6–38)}$$

For the AKF scatter plot the same model results in (rounded to the second decimal):

$$RMSE_{fit,AKF}(t_{PIDdelay}) = 1.33e^{5.34t_{PIDdelay}} + 5.25 * 10^{-13}e^{244t_{PIDdelay}}. \quad \textbf{Eq. (6–39)}$$

Solving the functions for the value defined by requirement F35 returns

$$t_{PIDdelay}(RMSE_{fit,CF} = 2.5\ [°]) = 0.074\ [\mathrm{s}], \quad \textbf{Eq. (6–40)}$$

for the P2PMAF+CF configuration and

$$t_{PIDdelay}(RMSE_{fit,AKF} = 2.5\ [°]) = 0.108\ [\mathrm{s}], \quad \textbf{Eq. (6–41)}$$

For the AKF configuration. Therefore, formulating the maximum allowed computational delay for the respective filter under reference setting.

6.3.3 Comparison and Sigma-Normalization

For computing the SNS for every factor, all previously discussed inputs are varied at the same time. The ranges of θ_{ES} and $t_{PIDdelay}$ were adapted to an approximately linear version around the reference values: $9.5\,° \leq \delta\theta_{ES} \leq 10.5\,°$ and $0.001\,s \leq \delta t_{PIDdelay} \leq 0.03\,s$. The maximum delay was chosen to be $t_{PIDdelay,max} = 0.03\,s$ since it equates to three times the reference value, which represents the worst-case assumption for scaling up to 3-DoF.

The response distributions for both filters are shown in Fig. 6-30 and Fig. 6-31. As the PDF, a Gamma distribution is used which allows to calculate $\sigma_{RMSEtot}$ for the SNS. It has the benefit that it reduces the effect of outliers, which resulted in system failure, on $\sigma_{RMSEtot}$. The characteristics of the Gamma distribution are explained in the appendix Section B.4.

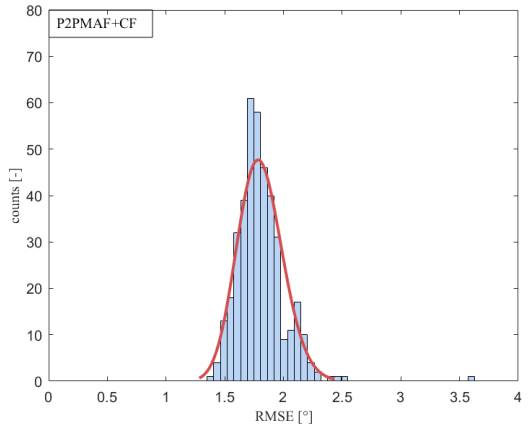


Fig. 6-30: Response Gamma distribution with $\sigma_{RMSEtot} = 0.192\,°$ and $\mu_{tot} = 1.804\,°$.

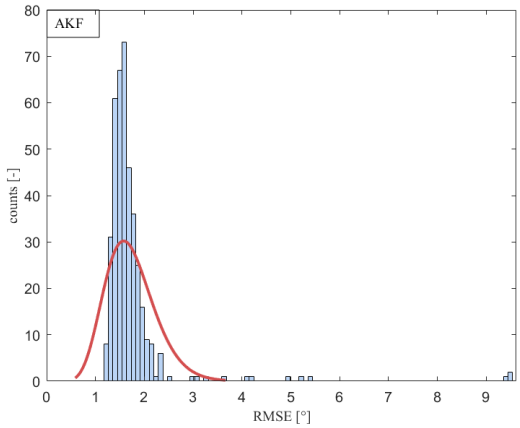


Fig. 6-31: Response Gamma distribution with $\sigma_{RMSEtot} = 0.510\,°$ and $\mu_{tot} = 1.727\,°$.

By accessing the Gamma distributions, the AKF provides a lower mean, while the P2PMAF+CF provides a lower standard deviation. However, the AKF response result is influenced by a larger quantity of outliers, which result from the control system failing to stabilize the pendulum. This is further emphasized by the CDFs, $c(RMSE)$, of both filters, which are shown below.

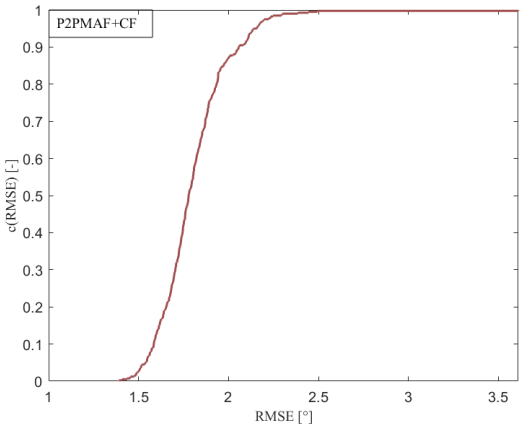


Fig. 6-32: CDF with $c(\mu_{tot}) = 0.5$.

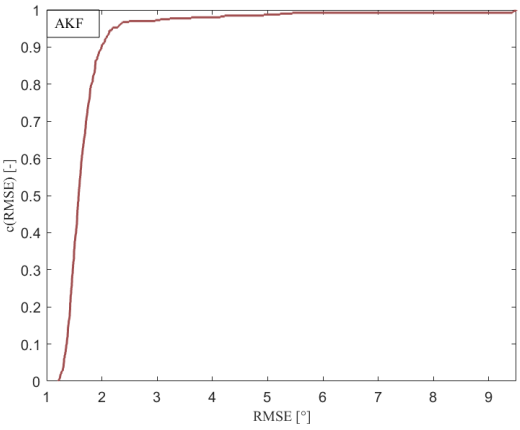


Fig. 6-33: CDF with $c(\mu_{tot}) = 0.5$.

It can be observed that $99.75\,\%$ of runs performed below the F35 limit for the P2PMAF+CF configuration, while for the AKF $97\,\%$ of runs performed below the F35 limit.

$$c_{RMSE,CF}(2.5\,[°]) = p(RMSE \leq 2.5\,[°]) = 99.75\,[\%] \qquad \textbf{Eq. (6–42)}$$

$$c_{RMSE,AKF}(2.5\,[°]) = p(RMSE \leq 2.5\,[°] = 97\,[\%] \qquad \textbf{Eq. (6–43)}$$

The true trajectories of all runs for both filters were plotted in Fig. 6-34: All 400 true trajectories for varying all key input factors with the given distributions. Fig. 6-34 and Fig. 6-35. The figures reveal the outlier trajectories, were the RCS could not lift the pendulum or failed to stabilize it.

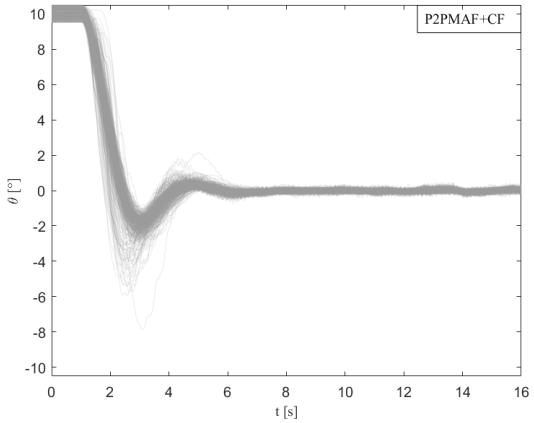


Fig. 6-34: All 400 true trajectories for varying all key input factors with the given distributions.

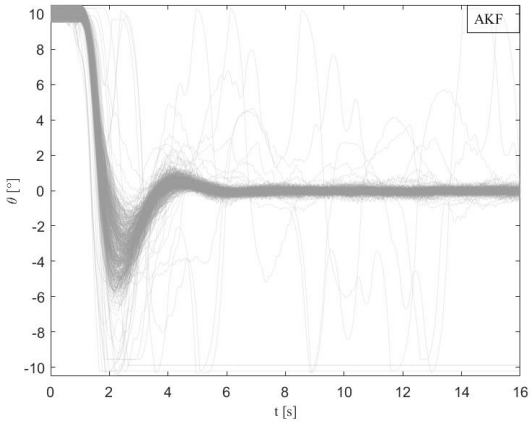


Fig. 6-35: All 400 true trajectories for varying all key input factors with the given distributions.

Addressing Q1 (see Chapter 3), it was decided that the 2σ-error in the response shall lie within the F35 boundary of 2.5 °, for the algorithm to be deemed reliable enough.

For the P2PMAF+CF configuration the 3σ-error in the response lies within the F35 performance boundary. Thus, qualifying the algorithm to be reliable enough in stabilizing the system.

For the AKF the 2σ-error in the response lies within the F35 performance boundary. Thus, qualifying the algorithm to be reliable enough in stabilizing the system. Additionally, the AKF algorithm leaves room for optimization of its Q, R and P_0 values.

To address Q2 the chosen key factors shall be revised and ranked. Key factors will be ranked regarding the filter configuration, since each possesses different reference response. The factors will by ranked by the squared SNS (SNS^2) value.

The cells of the tables below were color-coded to highlight the four highest SNS^2 and PDS values. The higher the respective value, the darker the color. A bluish color palette was chosen for the P2PMAF+CF configuration and a brownish for the AKF configuration.

Table 6-9 lists the normally distributed input factors while Table 6-10 lists the equally distributed input factors:

Table 6-9: Output statistics, PDS and squared SNS of the two filter configurations for the normally distributed inputs.

Input	Filter	Range	Output STD	Output Mean	PDS	SNS²
δt_{st}	P2PMAF+CF	30 [%]	0.096 [°]	1.589 [°]	0.006 [–]	0.0003 [–]
	AKF		0.060 [°]	1.347 [°]	0.018 [–]	0.0002 [–]
δm_0	P2PMAF+CF	3.56 [%]	0.103 [°]	1.629 [°]	0.810 [–]	0.0611 [–]
	AKF		0.073 [°]	1.383 [°]	0.916 [–]	0.0076 [–]
$\delta M_{SF,G,yy}$	P2PMAF+CF	3 [%]	0.086 [°]	1.648 [°]	0.094 [–]	0.0006 [–]
	AKF		0.062 [°]	1.383 [°]	−0.354 [–]	0.0008 [–]
$\delta M_{SF,A,xx}$	P2PMAF+CF		0.117 [°]	1.641[°]	0.553 [–]	0.0202 [–]
$\delta M_{SF,A,zz}$					−0.874 [–]	0.0503 [–]
$\delta M_{SF,A,xx}$	AKF		0.060 [°]	1.393 [°]	−0.111 [–]	0.0001 [–]
$\delta M_{SF,A,zz}$					−0.026 [–]	0.0000 [–]

Table 6-10: PDS and squared SNS of the two filter configurations for the equally distributed inputs.

Input	Filter	Min	Max	Input STD	PDS	SNS²
$\delta\theta_{ES}$	P2PMAF+CF	9.5 [°]	10.5 [°]	0.289 [°]	1.494 [–]	0.1362 [–]
	AKF				1.411 [–]	0.0171 [–]
$\delta\sigma_{G,y}$	P2PMAF+CF	$0.1\ \left[\frac{°}{s}\right]$	$0.5\ \left[\frac{°}{s}\right]$	$0.116\ \left[\frac{°}{s}\right]$	0.035 [–]	0.0726 [–]
	AKF				0.072 [–]	0.0301 [–]
$\delta\sigma_{A,x}$	P2PMAF+CF	$0.01\ \left[\frac{m}{s^2}\right]$	$0.1\ \left[\frac{m}{s^2}\right]$	$0.026\ \left[\frac{m}{s^2}\right]$	−0.001 [–]	0.0002 [–]
$\delta\sigma_{A,z}$					−0.013 [–]	0.0075 [–]
$\delta\sigma_{A,x}$	AKF				−0.017 [–]	0.0107 [–]
$\delta\sigma_{A,z}$					−0.068 [–]	0.0193 [–]
$\delta t_{PIDdelay}$	P2PMAF+CF	0.001 [s]	0.03 [s]	0.008 [s]	0.042 [–]	0.0904 [–]
	AKF				0.055 [–]	0.0151 [–]

Thus, for the P2PMAF+CF configuration the factor which the system has the highest sensitivity towards, is the end stop angle, followed by the computational delay, the gyroscope noise, and the total weight. These factors possess the largest share in the

overall variance of the system. However, the sum of all SNS^2 values does not equate to one:

$$\sum_i \left(S_{x_i}^{\sigma}\right)^2 = 0.4392. \qquad \textbf{Eq. (6–44)}$$

Therefore, it can be assumed, that the actual response function is not a linear combination of the inputs. Hence, most of the variance in the response may stem from nonlinear dependencies and couplings between the factors [70].

For the AKF the factor the factor which the system has the highest SNS towards, is the gyroscope noise, followed by the accelerometer noise on the z-axis, the end stop angle, and the computational delay. However, the sum of all SNS^2 values is even smaller compared to Eq. (6–44):

$$\sum_i \left(S_{x_i}^{\sigma}\right)^2 = 0.1011 \qquad \textbf{Eq. (6–45)}$$

Again, the actual response function is not a linear combination of the inputs. Hence, an even higher percentage of the variance in the response may stem from nonlinear dependencies and couplings between the factors as compared to the P2PMAF+CF [70].

These results conclude the characterization of the two filters, within the scope of the work. The next chapter will relate them to the research questions and scrutinize the validity of the approach and the utilized methods.

7 Discussion of the Results

In the following sections the fulfilment of the posed research questions, and the chosen methods to answer them, will be discussed and questioned. Therefore, this chapter refers to RS9 of the proposed designed experiment steps.

7.1 Tuning Guidelines and Design Insights

Placing Q1 in the context of established findings, it can be paraphrased to:

Q1: Which filter design allows for the 2σ-error in the response of the given system under changing key input factors to lie within the boundary defined by F35?

An AKF and CF, expanded with the P2PMAF, can both be implemented to reliably fulfill F35. Whereby the AKF appears to have the better mean performance μ_{tot}, while the P2PMAF+CF demonstrates a lower variance $(\sigma_{RMSEtot})^2$ in the response. Therefore, given that the implementation of the P2PMAF on the microcontroller is unproblematic, the latter configuration provides the more robust solution. A possible reason could be that the P2PMAF averages out noise on the accelerometer which was one of the major performance degrading factors on the AKF.

Major performance degrading factors are labeled key factors which shall be evaluated for Q2:

Q2: What are the key factors influencing the control system?

The key factors were found through categorization, preliminary tests and the SNS^2 value for each filter. The effect of the actuator switching time, t_{st}, and the scale factor, $\boldsymbol{M}_{SF}$, may have a prominent PDS, however, SNS^2 analysis diminishes their importance.

The P2PMAF+CF seems to be more sensitive towards external key factors like the end stop angle, the weight or the computational delay, while the AKF is more sensitive towards error-related factors, especially on the gyroscope, like the characteristic noise parameter on the y-axis $\sigma_{G,y}$. One possible explanation could be that the AKF will often neglect the measurement innovation y_k, therefore increasing the reliance on the gyroscope measurement.

Q3: Which rules apply to tuning the control system?

Boundaries: It was important to the hardware design team and the software implementation team to find out maxima of the end stop angle and the overall weight, for which the requirements are still fulfilled. Assuming reference values the maximum values are: $m_{0,max,CF} = 5.963\ kg$; $m_{0,max,AKF} = 6.185$ kg; $\theta_{ES,max,CF} = 11.680$ °; $\theta_{ES,max,AKF} = 11.878$ °.

Tuning rules: The best-practice tuning rules formulated in 5.4 have proven to produce reliable performance results in the evaluation and, therefore, will be recommended for implementation on the Teensy 3.6. For the AKF the tuning constant C was introduced to leave room for optimization. In simulation runs outside of the main test series it was varied in order to reduce the outliers in the response. Decreasing the tuning constant to $C = 0.5$ results in an improvement in the mean performance to $\mu_{tot} = 1.660$ °, an improvement in the STD to $\sigma_{RMSEtot} = 0.412$ °, and reduces the number of outliers so that $c_{RMSE,AKF}(2.5\ °) = 98.25\ \%$. However, it also increases the dependency on the

gyroscope measurement even further, which results in $\left(S^{\sigma}_{\sigma_{G,y}}\right)^2 = 0.0462$. This is evident since Q is proportional to C. Figures are shown in the appendix Section E.9.

With this, all research questions have been addressed. In contrast to the preliminary test results (see 6.2.2 and E.1), the P2PMAF+CF proved to be a more robust alternative to the AKF. However, the AKF leaves more room for optimization. Further, evaluating internal factors like the sensor sample rate, f_{sensor}, or the actuator frequency, f_{bb}, may reveal further insights about the behaviour of the control system. It was found out that the control system performance decreases vastly if computational delay values are not multitudes of $1/f_{sensor}$. However, this needs further research and could be a simulation error, resulting from the variable step solver.

The utilized methods for modelling, testing and evaluation may diminish the significance of the obtained results, which is discussed in the next section.

7.2 Methodological Shortcomings

The basic thought behind the test stand is that its dynamics resemble the thrustless instable flight condition of the WARR EX-4 upper stage during its coasting phase. If this assumption is too farfetched, the current progress of the project is in question. However, the filtered sensor measurements provide full-state feedback to the control structure, without utilizing a system model of the pendulum. Therefore, the filter algorithms should be applicable to any structure for attitude estimation, if gyroscope and accelerometer measurements are available.

For fast results at low expenses it was decided to focus this work on a simulation-only test environment for the project phases P0 and P1. However, a SIL simulation – with the control and filter algorithms implemented on the Teensy 3.6 being part of the loop – is necessary to accurately parse the effect of computational delay and other microcontroller related errors in the evaluation. This might be a future task of the software implementation team.

Model fidelity of the inverse pendulum was not the focus of this work. However, omitting damping and friction effects completely might have created a simulation model of the pendulum more instable than the real one, which might have complicated the stabilization task. This fact, combined with a more optimal control approach, such as a linear quadratic regulator targeted by [29], and a mechanical damping system [50] might make the filter obsolete. However, this is speculation, which will further be investigated by [30], which will also incorporate the 3-DoF pendulum model with higher fidelity. Other model inaccuracies include: the actuator model which assumes a symmetrical force output vector, which is not always the case according to [5], and the sensor error model, which can be improved by a proper Allan variance analysis on an optical table.

Lastly, the sum of the SNS^2 values resulted in values far smaller than 1. This shows that the major share of variance in the response, especially for the AKF, results from non-additive, higher-order relations and couplings not described in this work. Whether these effects impact the proposed ranking of key factors is uncertain.

8 Conclusion and Outlook

This thesis was designated to design a filter which could estimate the attitude of an inverse pendulum that aims to recreate the dynamics of suborbital rockets upper stage during instable flight. The filter was required to determine the attitude accurately enough to control the instable system via a cold gas RCS, using consumer-grade inertial sensors. Two filter algorithms were developed and tested in a simulation-only environment consisting of mathematical models of all relevant RCS subsystems. The peak-to-peak moving average filter was developed to prefilter the accelerometer output and feed it, together with the gyroscope measurements, to a complementary filter for state estimation. As a second filter algorithm, a simplified version of an adaptive Kalman filter was designed. The RMSE of the true trajectory to the commanded trajectory has served as the performance quantity to evaluate the filter algorithms. To facilitate the tuning and implementation process of the filters on the microcontroller and to reveal the limits of the RCS, the filter algorithms were characterized by conducting a designed Monte Carlo simulation experiment. Through categorization and the use of the squared sigma-normalized sensitivity, the key performance impairing factors were determined for each filter. Hence, revealing that the P2PMAF+CF is more susceptible to external factors, like the maximum angular deflection and computational delay, while the AKF is more susceptible to error-related factors, like the gyroscope noise. This shall help evaluate future RCS design decisions in the project. Overall, the AKF provides a better mean performance with more optimization potential, while the P2PMAF+CF is the more robust filter solution.

As a next step in project Exosphere, the control algorithms can be implemented on the Teensy 3.6 microcontroller to expand the Simulink model to a software-in-the-loop simulation environment. This may help incorporate microcontroller related issues not covered in the simulation-only approach.

Further, preliminary tests reveal that the sensor frequency and the actuator frequency could potentially be further reduced, while still maintaining the same performance. However, the in-built delays result in failed stabilization attempts for smaller sensor frequencies than the default one. Whether this a simulation issue or not may be resolved by including the microcontroller in the loop, creating the previously mentioned SIL testing environment.

Additionally, an alternative actuator solenoid valve, which allows to produce a higher thrust, can be evaluated with the tools created in this thesis to assess possible performance improvements.

At the time of typing this thesis, the communications module and the electronic power supply circuitry were about to be finalized, hence, completing the RCS in its P0 version.

Matthias Beck aims to advance the 3-DoF model and optimize the control structure for later project phases within his semester thesis [30]. Since the controller will be provided with full-state feedback, a linear quadratic regulator, as proposed by [29], may be a viable option.

A References

[1] Eric Berger, *Blue Origin just validated the new space movement.* [Online] Available: https://arstechnica.com/science/2016/10/blue-origin-just-validated-the-new-space-movement/. Accessed on: Jan. 12 2019.

[2] Doncaster Bill, Williams Caleb, and Shulman Jordan, “Nano/Microsatellite Market Forecast,” 2017. [Online] Available: http://www.spaceworkscommercial.com/wp-content/uploads/2018/01/SpaceWorks_Nano_Microsatellite_Market_Forecast_2017.pdf. Accessed on: Nov. 24 2018.

[3] d. Dy, Perrot Yann, and Pradal Robin, “Micro-launchers: what is the market?: Quick and flexible delivery of small payloads,” 2017.

[4] YOSHIFUMI INATANI and HIROHITO OHTSUKA, “SS-520 Nano satellite launcher and its flight result,” *32nd Annual AIAA/USU Conference on Small Satellites,* https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=4120&context=smallsat, 2018.

[5] Till Niklas Assmann, “Konzeption und Konstruktion eines Prototypen einer Lageregelungsaktuatorik mit Steuerdüsen für ine Suborbitalrakete,” Semester Thesis, Institute of Astronautics, Technical University of Munich, Munich, 2018 unpublished.

[6] G. P. Sutton and O. Biblarz, *Rocket propulsion elements,* 8th ed. Hoboken, N.J: Wiley, 2010.

[7] D. Gibbon, M. Paul, P. Smith, and R. McLellan, “The use of liquefied gases in small satellite propulsion systems,” in *37th Joint Propulsion Conference and Exhibit*, Salt Lake City,UT,U.S.A, 07082001.

[8] Malyuta, D., Collaud, X., Martins Gaspar, M., Rouaze, G. M. P., Pictet, R., Ivanov, A., & Mullin, N., “Active Model Rocket Stabilization Vias Cold Gas Thruster,” https://infoscience.epfl.ch/record/217915/files/I407-Malyuta.pdf, 2015.

[9] J. Ettl and J. Turner, “SHEFEX II - Precession Control System,” in *SpaceOps Conferences: SpaceOps 2014 Conference*, Pasadena, CA, 2014.

[10] J. Hadden, *Falcon 9 Users Guide rev02.* [Online] Available: https://www.spacex.com/sites/spacex/files/falcon_9_users_guide_rev_2.0.pdf. Accessed on: Jan. 13 2019.

[11] PRIME Faraday Partnership, “An Introduction to MEMS (Micro-electromechanical Systems),” *Prime Faraday Technology Watch,* https://www.lboro.ac.uk/microsites/mechman/research/ipm-ktn/pdf/Technology_review/an-introduction-to-mems.pdf, 2002.

[12] P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*: Artech House, 2013.

[13] J. K. Bekkeng, “Prototype Development of a Low-Cost Sounding Rocket Attitude Determination System and an Electric Field Instrument,” PhD thesis, Department of Physics, University of Oslo, Oslo, 2007.

[14] InvenSense Inc., Ed., “MPU-6000 and MPU-6050 Product Specification: Revision 3.4,” 2013. [Online] Available: www.invensense.com.

[15] ElectronicWings, *MPU6050 (Gyroscope + Accelerometer + Temperature) Sensor Module |…* [Online] Available: http://www.electronicwings.com/sensors-modules/mpu6050-gyroscope-accelerometer-temperature-sensor-module. Accessed on: Jan. 13 2019.

[16] R. Levy, D. Janiaud, J. Guerard, R. Taibi, and O. Le Traon, “A 50 nano-g resolution quartz Vibrating Beam Accelerometer,” in *2014 International Symposium on Inertial Sensors and Systems (ISISS)*, Laguna Beach, CA, USA, Feb. 2014 - Feb. 2014, pp. 1–4.

[17] M. Ahmadian, K. Jafari, and M. J. Sharifi, “Novel graphene-based optical MEMS accelerometer dependent on intensity modulation,” *ETRI Journal*, vol. 40, no. 6, pp. 794–801, 2018.

[18] B. Wu, Y. Yu, J. Xiong, and X. Zhang, “Silicon Integrated Interferometric Optical Gyroscope,” *Scientific Reports*, vol. 8, no. 1, p. 8766, https://doi.org/10.1038/s41598-018-27077-x, 2018.

[19] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel, “A Complementary Filter for Attitude Estimation of a Fixed-Wing UAV,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008: IROS 2008 ; 22-26 Sept., 2008, Acropolis Convention Center, Nice, France*, Nice, 2008, pp. 340–345.

[20] K. Akcay, “Performance Metrics for Estimation Filters,” Master Thesis, MIDDLE EAST TECHNICAL UNIVERSITY, 2005.

[21] N. K. *et al.,* “Noise modeling and analysis of an IMU-based attitude sensor: improvement of performance by filtering and sensor fusion,” in *Advances in Optical and Mechanical Technologies for Telescopes and Instrumentation II*, Edinburgh, United Kingdom, 2016, 99126W.

[22] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*. New York, NY: Springer, 2014.

[23] Festo AG & Co. KG, Ed., “Magnetventil MHE4-MS1H-3/2G-1/4-K: Datenblatt,” 2013.

[24] Sparkfun, *Teensy 3.6 - DEV-14057 - SparkFun Electronics.* [Online] Available: https://www.sparkfun.com/products/14057. Accessed on: Dec. 14 2018.

[25] Core Electronics, *Teensy 3.5 & 3.6 Review - Tutorial.* [Online] Available: https://core-electronics.com.au/tutorials/teensy-3.5-and-3.6-review.html. Accessed on: Dec. 14 2018.

[26] Till Niklas Assmann, "Project Meeting on Actuators", Face-to-Face meeting, Sep. 2018.

[27] Till Assmann, Flavio Rehn, and Tim Klose, “Project Exosphere Preliminary Requirements,” Nov. 2018.

[28] *DETAIL SPECIFICATION - FLIGHT CONTROL SYSTEMS - DESIGN, INSTALLATION AND TEST OF*, MIL-F 9490D, 1975.

[29] E. S. Schmidt, J. Louke, K. Amell, J. Hickman, and B. Wiles, “Development of a Low-Cost, Open Hardware Attitude Control System for High Powered Rockets,” in *AIAA SPACE 2015 Conference and Exposition*, Pasadena, California, 2015, p. 157.

[30] Matthias Beck, “Controller Design of a Reaction Control System for Suborbital Rockets Using Cold Gas Thrusters,” Semester Thesis, Institute of Flight System Dynamics, Technical University of Munich, Munich, 2019 unpublished.

[31] Matthias Beck, "Project Meeting on 3D Model and Potential Control Algorithms", Face-to-Face meeting, Oct. 2018.

[32] Tim Klose, "Project Meeting on microcontroller and IMU", Face-to-Face meeting, Oct. 2018.

[33] The MathWorks Inc., *MAAB Control Algorithm Modeling.* [Online] Available: https://de.mathworks.com/help/simulink/maab-control-algorithm-modeling.html. Accessed on: Oct. 23 2018.

[34] Rick Poorman, *Simulink Style Guide Version 1.0.* Accessed on: Oct. 25 2018.

[35] The MathWorks Inc., *Second-Order Integrator Limited.* [Online] Available: https://www.mathworks.com/help/simulink/slref/secondorderintegratorlimited.html . Accessed on: Dec. 10 2018.

[36] The MathWorks Inc., *Zero-Crossing Detection.* [Online] Available: https://de.mathworks.com/help/simulink/ug/zero-crossing-detection.html. Accessed on: Oct. 23 2018.

[37] A. M, “PID Control,” http://www.eolss.net/ebooks/Sample%20Chapters/C18/E6-43-03-03.pdf.

[38] The MathWorks Inc., *PID Tuner.* [Online] Available: https://www.mathworks.com/help/control/ref/pidtuner-app.html. Accessed on: Dec. 12 2018.

[39] The MathWorks Inc., *pidtune: PID tuning algorithm for linear plant model.* [Online] Available: https://www.mathworks.com/help/control/ref/pidtune.html. Accessed on: Dec. 12 2018.

[40] D. Liberzon, *Calculus of variations and optimal control theory: A concise introduction*. Princeton, NJ: Princeton Univ. Press, 2012.

[41] F. W. J. Olver, *NIST handbook of mathematical functions: Companion to the digital library of mathematical functions*. Cambridge: Cambridge University Press, 2010.

[42] The MathWorks Inc., *Solve Stiff ODEs.* [Online] Available: https://de.mathworks.com/help/matlab/math/solve-stiff-odes.html. Accessed on: Dec. 12 2018.

[43] The MathWorks Inc., *Choose an ODE Solver.* [Online] Available: https://de.mathworks.com/help/matlab/math/choose-an-ode-solver.html. Accessed on: Dec. 12 2018.

[44] H. GREER and D. J. GRIEP, “Dynamic performance of low-thrust, cold-gas reaction jets in a vacuum,” *Journal of Spacecraft and Rockets*, vol. 4, no. 8, pp. 983–990, 1967.

[45] N. El-Sheimy, H. Hou, and X. Niu, “Analysis and Modeling of Inertial Sensors Using Allan Variance,” *IEEE Trans. Instrum. Meas.*, vol. 57, no. 1, pp. 140–149, 2008.

[46] X. Y. Dai, Z. G. Chen, and X. Xie, “The Gyro Random Walk Analysis Based on Allan Variance,” *AMM*, vol. 668-669, pp. 953–956, 2014.

[47] C. MacKinnon, “IMU Errors and Their Effects,” Feb. 2014.

[48] Warren S. Flenniken IV, John H. Wall, David M. Bevly, “Characterization of Various IMU Error Sources and the Effect on Navigation Performance,” 2005.

[49] Trusov Alexander A., “Overview of MEMS Gyroscopes:: History, Principles of Operations, Types of Measurements,” MicroSystems Laboratory, Mechanical and Aerospace Engineering, University of California, Irvine.

[50] Li, Z., Lao, M., Phang, S. K., Redhwan, M., Hamid, A., Tang, K. Z., & Lin, F., “Development and Design Methodology of an Anti-Vibration System on Micro-UAVs,” *International Micro Air Vehicle Conference and Flight Competition (IMAV))*, pp. 223-228, 2017.

[51] The MathWorks Inc., *FFT.* [Online] Available: https://de.mathworks.com/help/matlab/ref/fft.html?searchHighlight=fft&s_tid=doc_srchtitle. Accessed on: Dec. 03 2018.

[52] M.Sc. Eng. Hristo Zhivomirov, *Phase Difference Measurement with MATLAB Implementation.* [Online] Available: https://de.mathworks.com/matlabcentral/fileexchange/48025-phase-difference-measurement-with-matlab-implementation. Accessed on: Dec. 03 2018.

[53] The MathWorks Inc., *Smoothing Splines: About Smoothing Splines.* [Online] Available: https://de.mathworks.com/help/curvefit/smoothing-splines.html. Accessed on: Dec. 03 2018.

[54] Diganta Das and R. Roseline Mary, “Real Time Noise Filtering for Low Cost IMU Sensor,” *Asian Journal of Information Technology*, 563-543, 2017.

[55] Holzapfel Florian, “Flight Control 2: Lecture Notes,” Technical University of Munich, Munich, 2018.

[56] Messmann David, “Attitude Estimation for the MOVE-II CubeSat,” Master Thesis, Chair of Astronautics, Technical University of Munich, Munich, 2018.

[57] The MathWorks Inc., *Kalman Filter.* [Online] Available: https://de.mathworks.com/help/ident/ref/kalmanfilter.html?s_tid=doc_ta. Accessed on: Nov. 25 2018.

[58] R. Mehra, “Approaches to adaptive filtering,” *IEEE Trans. Automat. Contr.*, vol. 17, no. 5, pp. 693–698, 1972.

[59] A. H. Mohamed and K. P. Schwarz, “Adaptive Kalman Filtering for INS/GPS,” *Journal of Geodesy*, vol. 73, no. 4, pp. 193–203, 1999.

[60] J. Antony, *Design of experiments for engineers and scientists,* 2nd ed. London: Elsevier, 2014.

[61] John A. Sokolowski, Catherine M. Banks, Ed., *Modeling and Simulation Fundamentals: Theoretical Underpinnings and Practical Domains*: Wiley, 2010.

[62] G. E. P. Box and D. W. Behnken, “Some New Three Level Designs for the Study of Quantitative Variables,” *Technometrics*, vol. 2, no. 4, p. 455, 1960.

[63] The Pennsylvania State University, *Introduction to Design of Experiments: Steps for Planning, Conducting and Analyzing an Experiment.* [Online] Available: https://onlinecourses.science.psu.edu/stat503/node/7/. Accessed on: Nov. 17 2018.

[64] J. K. Telford, “A Brief Introduction to Design of Experiments,” *Johns Hopkins apl technical digest*, pp. 224–232, https://pdfs.semanticscholar.org/6891/51985c45239829200a17fe00fab9123c4466.pdf, 2007.

[65] X. R. Li and Z. Zhao, “Measures of performance for evaluation of estimators and filters,” in *Signal and Data Processing of Small Targets 2001*, San Diego, CA, USA, 2001, p. 530.

[66] J. Guo and M. Zhong, “Calibration and Compensation of the Scale Factor Errors in DTG POS,” *IEEE Trans. Instrum. Meas.*, vol. 62, no. 10, pp. 2784–2794, 2013.

[67] V. Czitrom, “One-Factor-at-a-Time versus Designed Experiments,” *The American Statistician*, vol. 53, no. 2, p. 126, 1999.

[68] Sven Erik Jørgensen, Brian D. Fath, “Fundamentals of ecological modelling: Application in environmental management and research,” *Developments in Environmental Modelling*, no. 23, pp. 2–399, 2011.

[69] D. M. Hamby, “A review of techniques for parameter sensitivity analysis of environmental models,” (eng), *Environmental monitoring and assessment*, vol. 32, no. 2, pp. 135–154, 1994.

[70] A. Saltelli *et al., Global Sensitivity Analysis. The Primer*. Chichester, UK: John Wiley & Sons, Ltd, 2007.

[71] Christian Walck, *Hand-book on Statistical Distribution for experimentalists*, 2007.

[72] D. D. Wackerly, W. Mendenhall, and R. L. Scheaffer, *Mathematical statistics with applications,* 7th ed. Pacific Grove, CA: Duxbury, 2008.

[73] Saul Stahl, “The Evolution of the Normal Distribution,” *Mathematics Magazine*, no. vol. 79, pp. 96–113, 2006.

[74] rocscience, *Uniform Distribution.* [Online] Available: https://www.rocscience.com/help/rs2/phase2_model/Uniform_Distribution.htm. Accessed on: Jan. 20 2019.

B Primer on Kalman Filter and Probability

This section of the appendix shall serve as an overview of basic principles of Kalman filtering and probability theory. It will help understand the theory behind both Chapters 5.3 and 6.

B.1 Kalman Filter Basics

The Kalman Filter is considered a Bayesian estimator because it is based on Bayes' theorem which can be formulated as

$$p(X|Y) = \frac{p(X \cap Y)}{p(Y)} = \frac{p(Y|X)p(X)}{p(Y)}, \; for\, p(Y) \neq 0. \qquad \textbf{Eq. (B–1)}$$

The rule describes the probability, p, of event X occurring given that event Y is true. This is used in a Kalman Filter to obtain the a-posteriori value, from the a-priori value and an observation z. In this sense, x^- corresponds to the a-priori value of x , while x^+ corresponds to the a-posteriori value of x [12]. Throughout the Kalman Filter theory part, Section 5.3.1, this concept will be further expanded upon.

Matrices such as the measurement covariance $\boldsymbol{R}$, the system covariance $\boldsymbol{Q}$ and the error covariance $\boldsymbol{P}$ will be reduced to their one-dimensional scalar counterpart, such as the measurement variance R, the system variance Q and the error variance P.

In general, the variance, σ^2, of a random variable $\boldsymbol{x}$ is the expected value of the squared deviation from the mean, μ, of $\boldsymbol{x}$:

$$\mu = E[\boldsymbol{x}] = \frac{1}{N}\sum_{i=1}^{N} x_i \,, \; with\, \boldsymbol{x} \in \mathbb{R}^{N\times 1}, \qquad \textbf{Eq. (B–2)}$$

$$\sigma^2 = E[(\boldsymbol{x} - \mu)].$$

Therefore, the standard deviation equates to

$$\sigma = \sqrt{E[(\boldsymbol{x} - \mu)]}. \qquad \textbf{Eq. (B–3)}$$

Hereby, the mean and the variance are considered the first and the second statistical moment respectively [71]. The skewness and the kurtosis of a distribution define the third and the fourth moment but will not be used for the evaluation in this thesis.

For performance evaluation of estimator the root-mean-square-error (RMSE) is recommended [72]:

$$RMSE = \sqrt{E[(\boldsymbol{x} - \hat{\boldsymbol{x}})^2]} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}. \qquad \textbf{Eq. (B–4)}$$

B.2 Gaussian Normal Distribution

The gaussian normal distribution will be used to describe the random input, x, of a Monte Carlo simulation, which is used for the evaluation process of the filter. The probability density function (PDF) of a normal distribution equates to:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \qquad \textbf{Eq. (B–5)}$$

Integrating the probability density function $f(x)$ over a interval of $-\infty$ to a random continuous input x reveals the cumulative density function (CDF) [71, 73]:

$$c(x) = \int_{-\infty}^{x} f(x)\,dx = \int_{-\infty}^{\sigma} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\,dx. \qquad \textbf{Eq. (B–6)}$$

Now, the probability , $p(x)$, of an event occurring within one standard deviation is calculated by

$$p(x) = \int_{-\sigma}^{\sigma} f(x)\,dx = \int_{-\sigma}^{\sigma} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\,dx = 0.6827, \qquad \textbf{Eq. (B–7)}$$

and accordingly, for two and three standard deviations by

$$p(x) = \int_{-2\sigma}^{2\sigma} f(x)\,dx = 0.9545, \qquad \textbf{Eq. (B–8)}$$

$$p(x) = \int_{-3\sigma}^{3\sigma} f(x)\,dx = 0.9973.$$

Reformulated, the CDF represents the probability that the random the random variable δx takes on a value less than or equal to x:

$$c_{\delta x}(x) = p(\delta x \leq x). \qquad \textbf{Eq. (B–9)}$$

These equations hold true for any distribution which can be described by a STD.

A randomly generated value within this distribution will be described by the mean and the standard deviation with the following notation for a random variable δx:

$$\delta x = \mathcal{N}[\mu, \sigma]. \qquad \textbf{Eq. (B–10)}$$

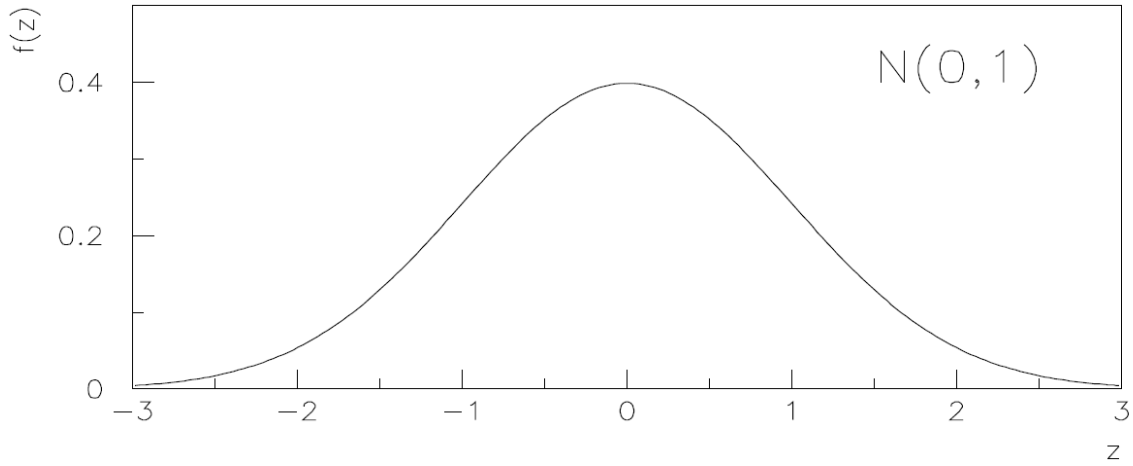


Fig. B-1: The standard normal distribution from [71] with the transformation $z = (x - \mu)/\sigma$.

Besides the gaussian, other PDFs will be used to approximate a distribution.

B.3 Equal Distribution

First, there is the equal also referred to as uniform distribution. The equal distribution is uniformly distributed and possess the PDF [71]:

$$f(x): \begin{cases} \frac{1}{x_{max} - x_{min}}, & for\ x_{min} \leq x \leq x_{max} \\ 0, & for\ x < x_{min} \\ 0, & for\ x > x_{max} \end{cases}. \qquad \textbf{Eq. (B–11)}$$

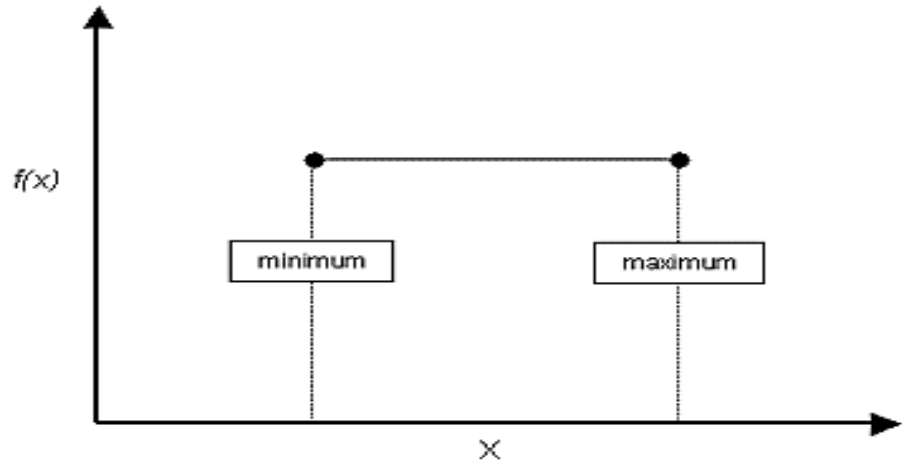


Fig. B-2: PDF of the equal distributions from [74].

An equally distributed arbitrary input will be described with

$$x = \mathcal{U}[x_{min}, x_{max}]. \qquad \textbf{Eq. (B–12)}$$

The expected value equates to

$$E[x] = \mu = \frac{x_{max}+x_{min}}{2}, \qquad \textbf{Eq. (B–13)}$$

and the standard deviation to

$$\sigma = \frac{x_{max}-x_{min}}{\sqrt{12}}. \qquad \textbf{Eq. (B–14)}$$

B.4 Gamma Distribution

In this work the Gamma distribution is used in the presence of positive outliers in the response. The PDF is [71]:

$$f(x, a, b) = \frac{a(ax)^{b-1}e^{-ax}}{\Gamma(b)}. \qquad \textbf{Eq. (B–15)}$$

Where the parameters a and b are positive real quantities as is the variable x. For $b \leq 1$ the distribution is J-shaped and for $b > 1$ it is unimodal with its maximum value at $x = (b-1)/a$.

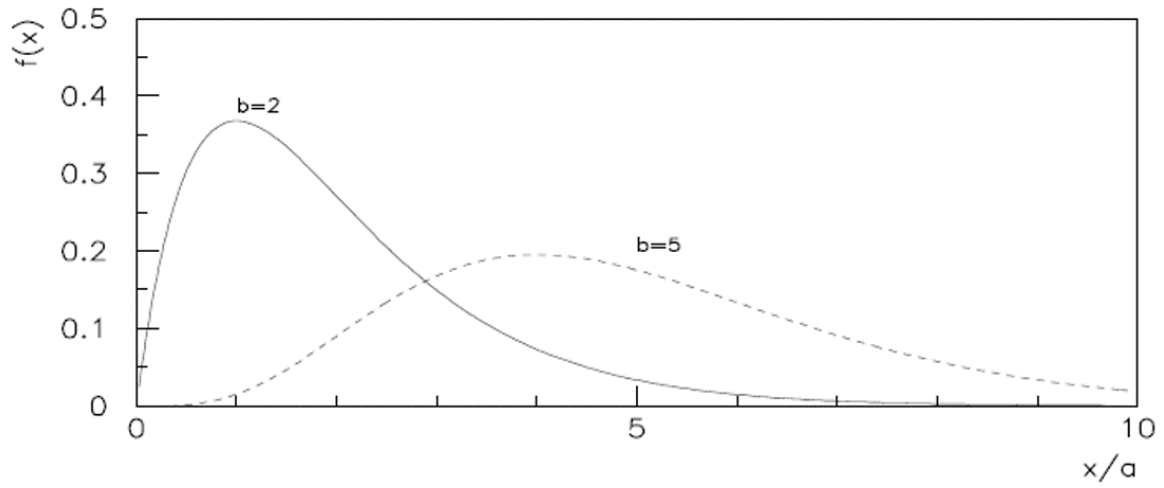


Fig. B-3: Examples of Gamma distributions adopted from [71].

The Gamma distribution is named after Euler's Gamma function $\Gamma(b)$:

$$\Gamma(x+1) = x!. \qquad \textbf{Eq. (B–16)}$$

A Gamma distributed input will be described with

$$x = \Gamma[a, b]. \qquad \textbf{Eq. (B–17)}$$

The first moment / expected value equates to

$$E[x] = \mu = \frac{b}{a}, \qquad \textbf{Eq. (B–18)}$$

and the standard deviation to

$$\sigma = \frac{\sqrt{b}}{a}. \qquad \textbf{Eq. (B–19)}$$

C How to Use the Simulink Model

This section presents a step-by-step guide on how to use the Simulink model explained in the Chapters 4 and 5.

To experience no portability issues, the user of the model should make sure to have the MATLAB version `Matlab R2017b SimulinkVersion 9.0` with the `Continuous Toolbox Control, Design Toolbox` and the `Control System Toolbox` installed.

C.1 Initialization

The most recent version of the Simulink model is the `System_8b_allfilters.slx` file. Before starting a simulation run, all parameters must be initialized:

Open the `initialize_RCS.m` MATLAB script and run the first section of it labelled `Initialize RCS Model`.

Now run the next section of the script labelled `Open Simulink Model` or manually set up the simulation time and the solver configuration in Simulink. To make sure that the same results are obtained as in this thesis, the solver must be set to the `Ode15`, which is not the default configuration for Simulink. The solver can be selected through the tab `Model Configuration Parameters`. Adjust the values to the ones shown in the figure below.

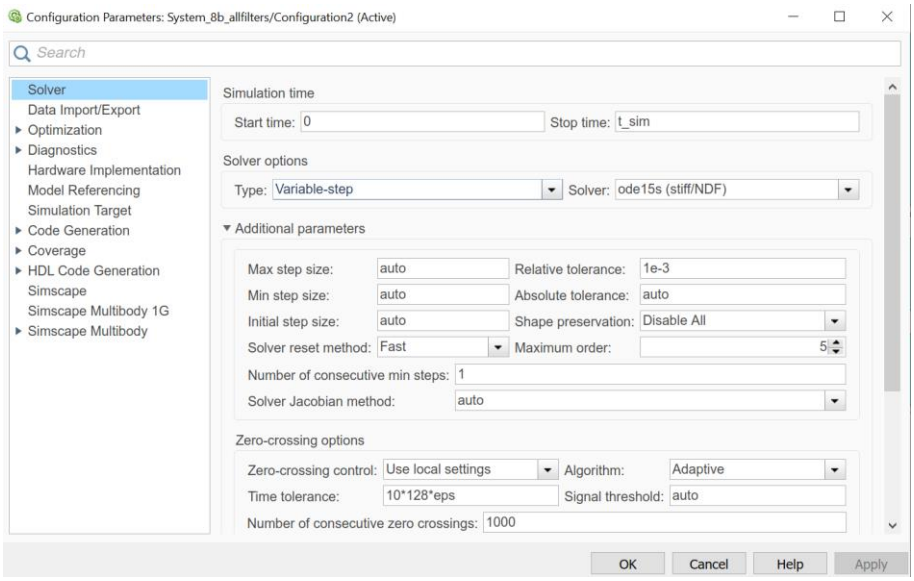


Fig. C-4: `Model Configuration Parameters` used in this work.

Note that the simulation runs on a variable step size. However, the sensor and filter blocks are sampled with the sensor sample rate, while the two-step controller is sampled with the actuator frequency.

Now that the simulation is initialized with the reference values of the parameters, the simulation can be run. Results are exported to MATLAB workspace to evaluate them. The last section of the `initialize_RCS.m` script allows for a quick evaluation of the results by computing the RMSE and the total gas consumption.

C.2 Orientation in the Model

Aside from the variables initialized through the script, some variables must be assigned through masks of the respective subsystem. A mask is a custom user interface for a block in Simulink. If a block possesses a mask, it will appear when double-clicking on it. A typical mask layout is presented in the figure below.

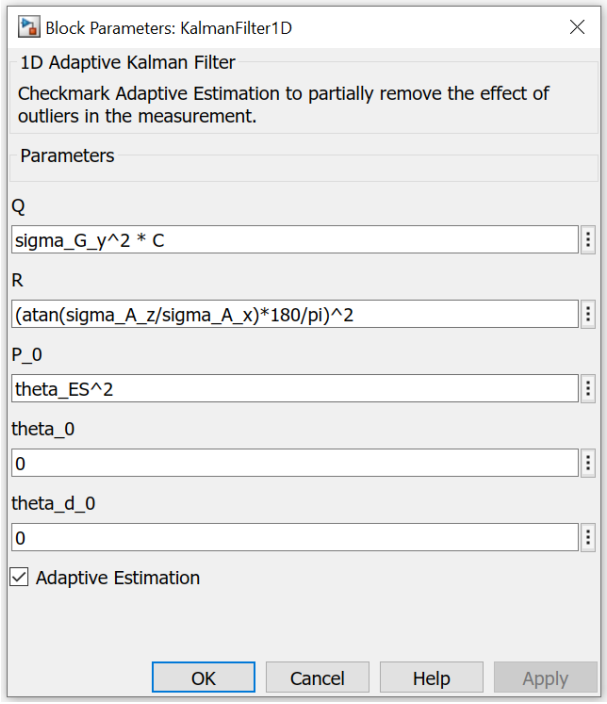


Fig. C-5: `KalmanFilter1D` mask with tuning rules implemented.

The mask will give the option to enter a value for the respective variable or checkmark a box to enable a specific function. In the case of Fig. C-5, checking the box would result in activation the innovation-based estimator.

To reveal the subsystem elements under the mask, the small arrow on the lower right corner must be clicked. Another way is to right-click and then search for `mask – look under mask`.

C.3 Customization

The user-friendly simulation interface was created to encourage experimenting with the model parameters. The result can be visualized at every line by connecting it to a scope or utilizing the already existing scopes. However, experimenting with the values may lead to system failure. An often-occurring true trajectory is either that the system becomes unstable or that the pendulum lies completely flat. This happens for entering a mass over the established boundary values without adjusting the PID gains.

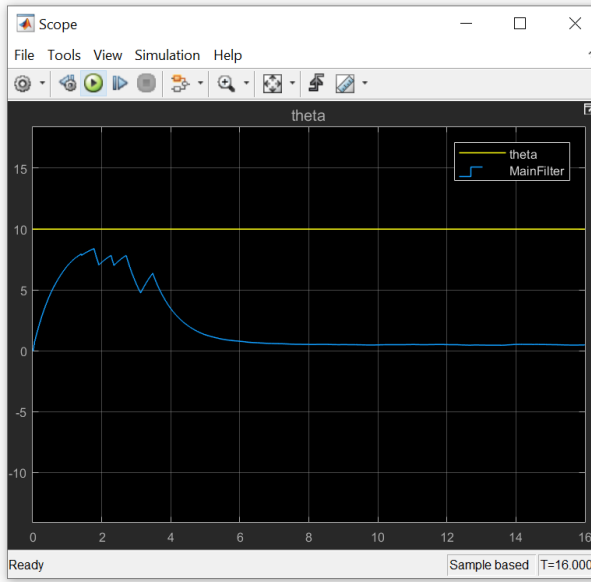


Fig. C-6: The control system fails to lift the pendulum for $m_0 = 6.5\ \mathrm{kg}$.

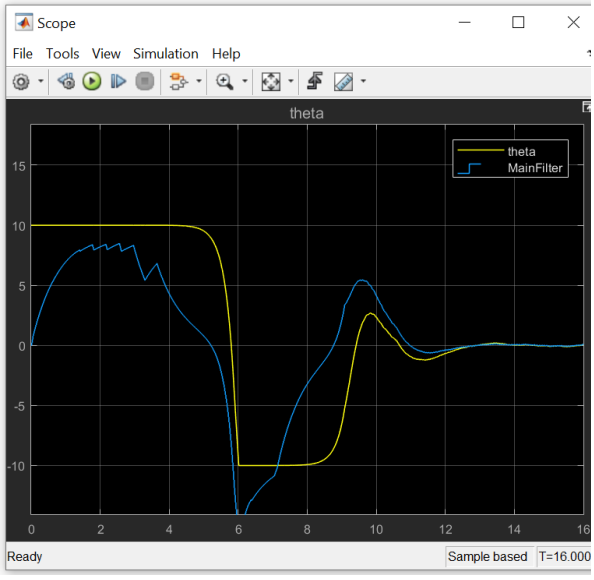


Fig. C-7: The system is initially unstable for $m_0 = 6.165\ \mathrm{kg}$.

If the system exhibits initially unstable behaviour, like shown in Fig. C-7, adjusting the PID gains in the reduced model may help. It is labelled `PID_gain_tuning_model.slx` and only contains the plant and the controller. However, for system failure as it is the case in Fig. C-6 adjusting the gains will not help.

C.4 Evaluation

Some data like the commanded trajectory, `theta_com`, or the true trajectory, `theta`, is already exported to the MATLAB workspace. The label discrete behind a name indicates that it uses the same sample time as the sensor. Continuous outputs will possess a varying sample size depending on the step size the solver chooses.

Multiple tools for evaluation were compiled. Most commonly the `eva.m` script was used compute and plot the values and figures in Section 6.3. It will iterate through the set number of runs L and output a histogram and a scatter plot with a linear regression function. The `eva3D.m` is used for creating the multivariate regression fits as for the scale factor or the accelerometer noise. Finally, the `SNStool.m` will compute the SNS^2 value for the input `delta_x_all_input.mat`, which consists out of the arbitrary inputs of all key factors.

To recreate the exact same results as in this work, it is recommended to use the saved arbitrary input generated for the main test series. Or directly input the corresponding RMSE values and then run the las section of the respective script without running the simulation.

D Simulink Subsystems

The section shows the subsystems described throughout the Chapter 4 and 5. This shall serve as an aid for orientation in the Simulink model `System_8b_allfilters.slx`.

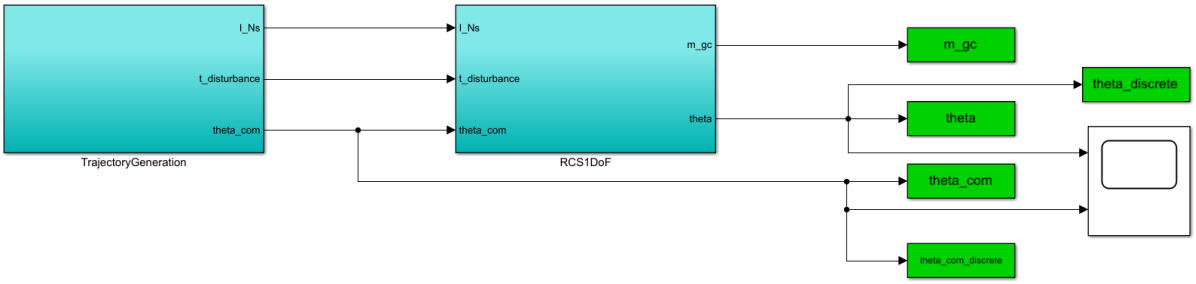


Fig. D-8: First level of the Simulink model.

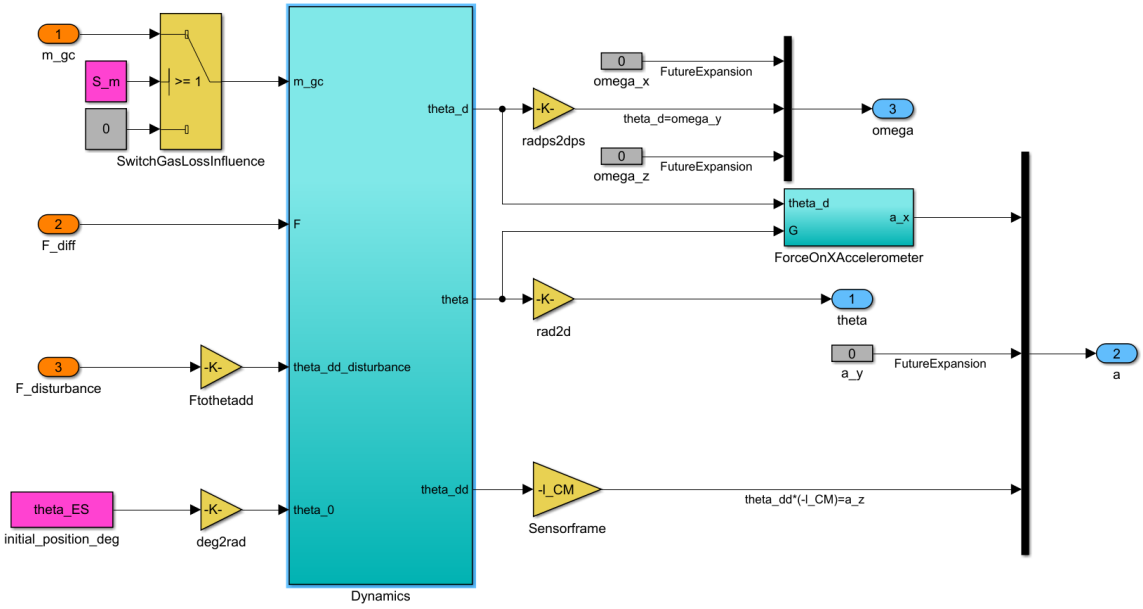


Fig. D-9: `InversePendulum1DoF` block with transformation to sensor frame and switch to account for mass due to gas burn.

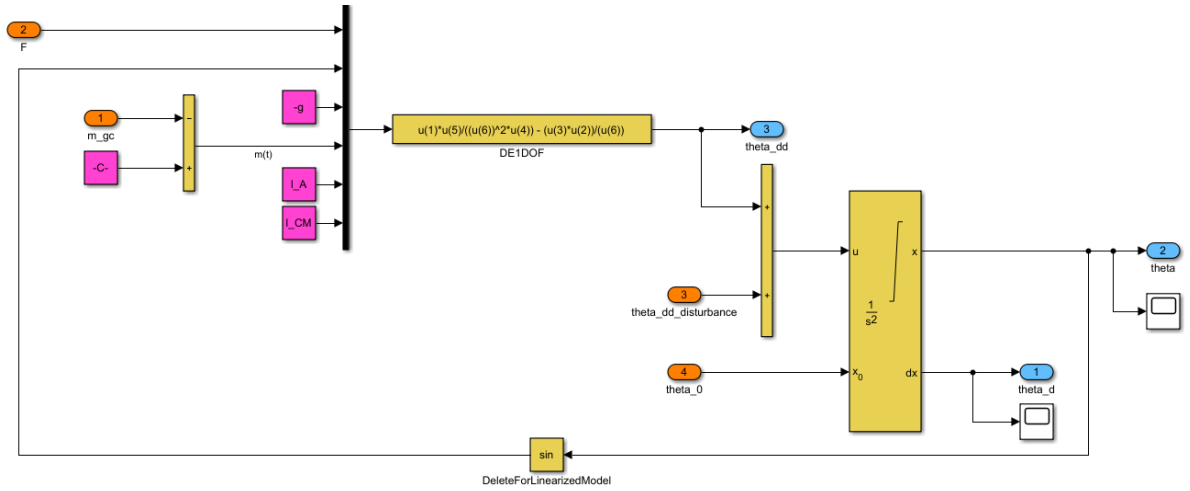


Fig. D-10: 1-DoF inverse pendulum `Dynamics` block with non-linear DE and `Integrator, Second-Order` block.

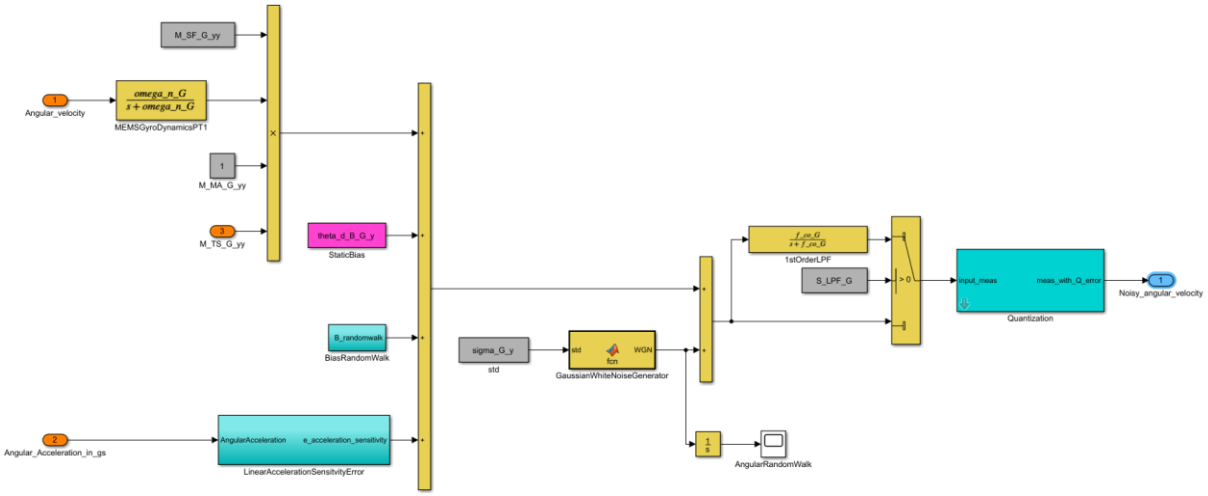


Fig. D-11: The `SensorModelGyroY` block.

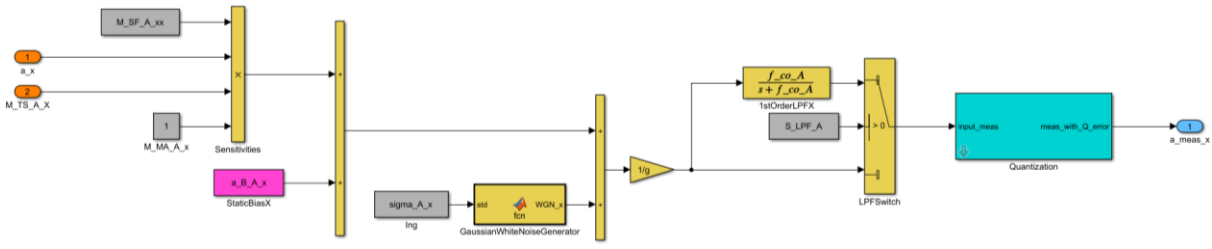


Fig. D-12: The `AccelerometerX` block. The `AccelerometerZ` block is similar.

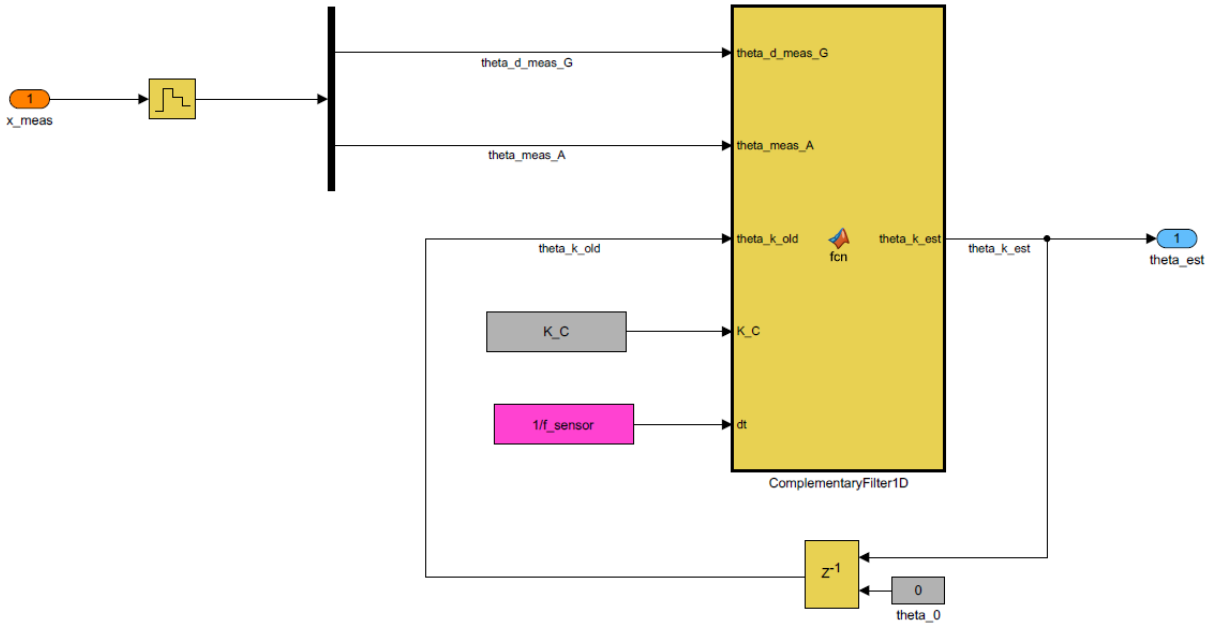


Fig. D-13: Complementary Filter Simulink Model

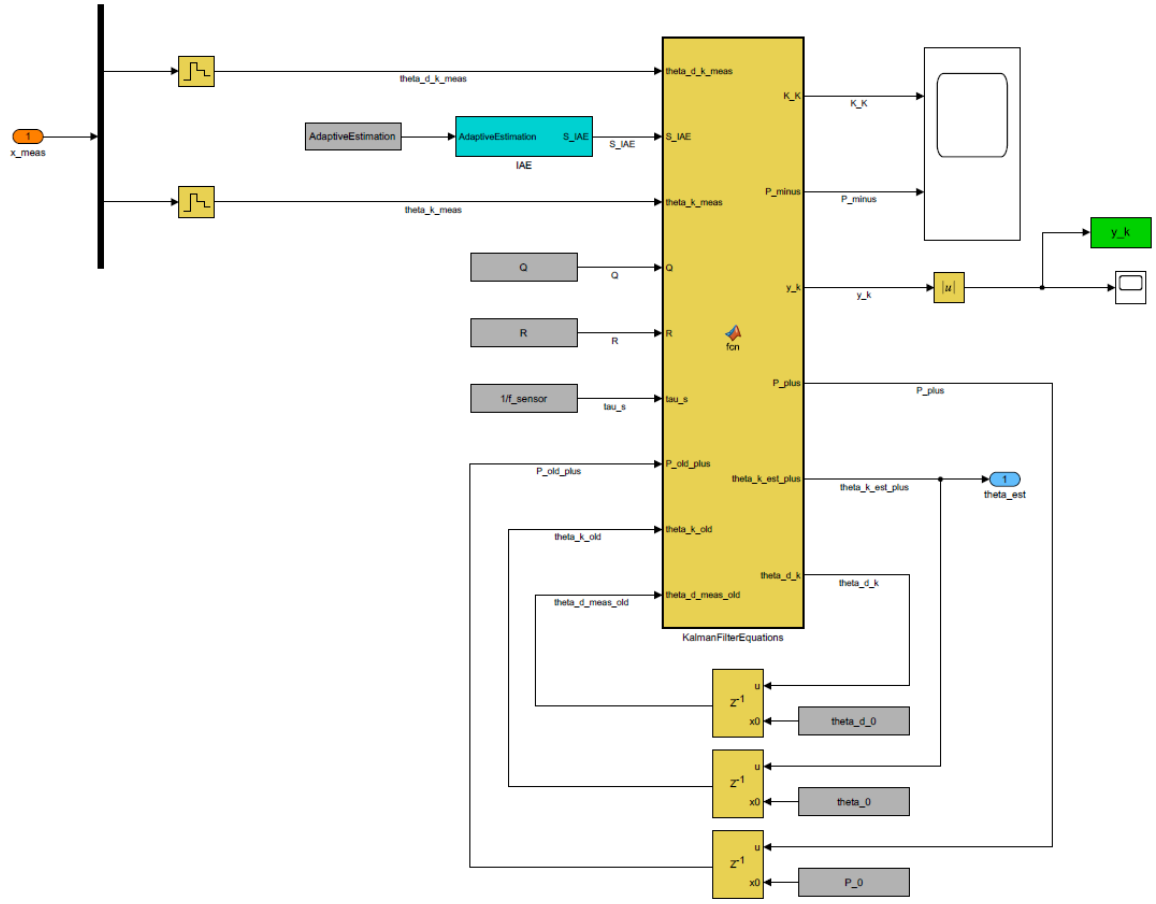


Fig. D-14: The Kalman filter with IAE-extension Simulink model.

E Additional Simulation Results

The section comprises all graphs which are of secondary interest for the evaluation. This includes the preliminary test results and $RMSE_{\hat{\theta}}$ scatter plots. To repeat the experiments, the input distributions and output distribution were saved for every tested factor.

E.1 Trajectories

This section shows the performance of the system for the other two established trajectories, $T_{mode} = 1$ and $T_{mode} = 2$.

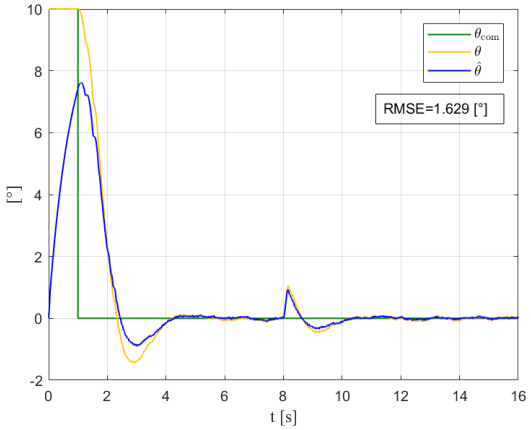


Fig. E-15: $T_{mode} = 1$ for the P2PMAF+CF configuration.

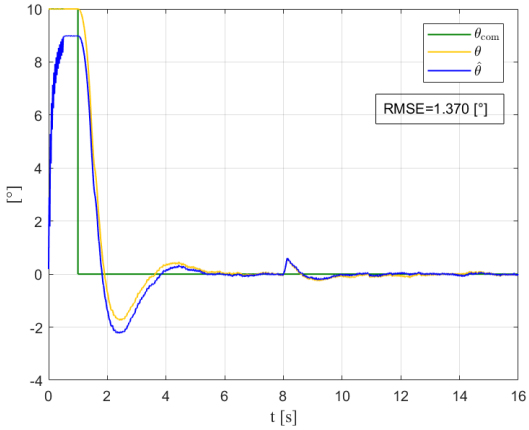


Fig. E-16: $T_{mode} = 1$ for the AKF configuration.

The AKF shows a better performance for the $T_{mode} = 1$. However, both filters manage to stabilize the pendulum after a disturbance.

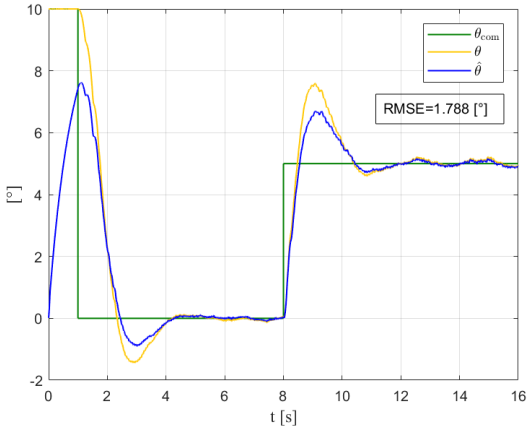


Fig. E-17: $T_{mode} = 2$ for the P2PMAF+CF configuration.

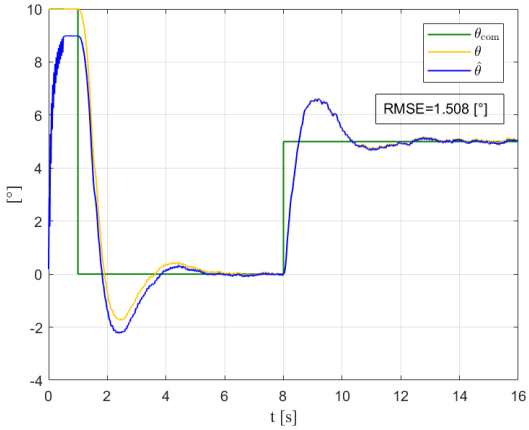


Fig. E-18: $T_{mode} = 2$ for the AKF configuration.

For $T_{mode} = 2$, an advantage of the AKF becomes evident. Once the Kalman gain converges towards the optimal value, the estimation accuracy surpasses the P2PMAF+CF configuration which does not change its gain.

E.2 Quantization Error

The quantization is a random error process which all sensors exhibit. It depends on the available bit size, B, of the IMU and the measurement range. A measurement range of $a_{range} = 4\,\mathrm{g}$ – with the minimum measurable value being $-2\,\mathrm{g}$ and the maximum being $2\,\mathrm{g}$ – on the accelerometer results in the quantization error shown in Fig. E-19.

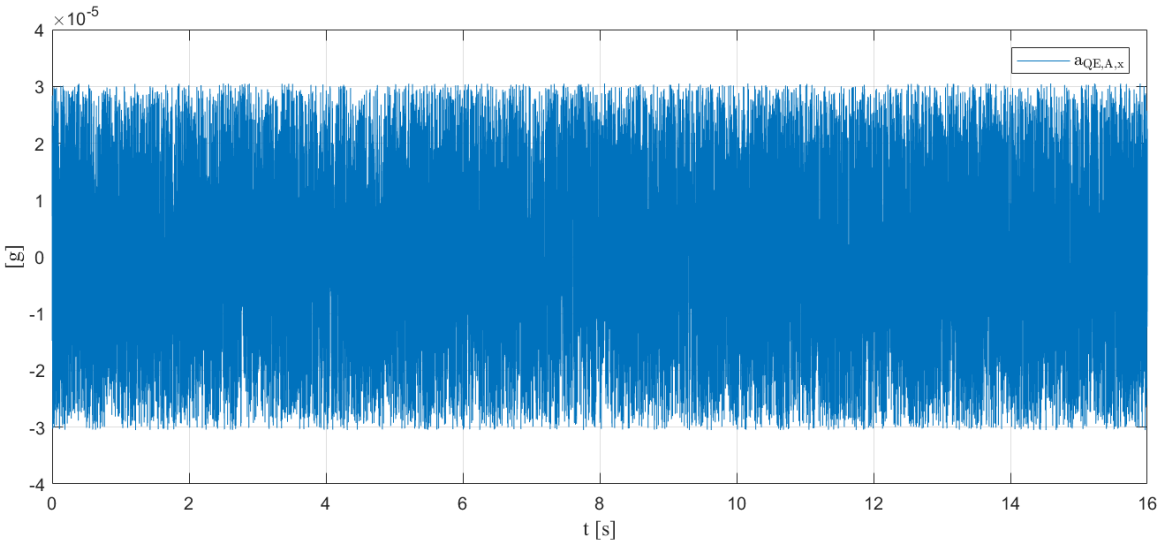


Fig. E-19: Quantization error resulting from rounding on the x-axis of the accelerometer.

At its peak the quantization error reaches $a_{QE,A,x} = 30\,\mu\mathrm{g}$, which not enough to further analyze it in the main test series. If the measurement range on the accelerometer is increased to the maximum of $a_{range} = 32\,\mathrm{g}$, the maximum quantization error increases to $a_{QE,A,x} = 244\,\mu\mathrm{g}$. This was not enough the influence the performance of the control system. Therefore, the quantization error will be labeled a nuisance factor.

E.3 Environmental Temperature

For the environmental temperature no noticeable performance degradation could be identified in the preliminary test series for the AKF. It will not be further analyzed.

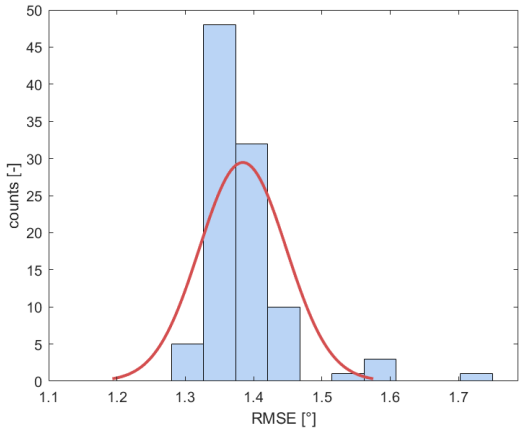


Fig. E-20: AKF histogram for preliminary test result of T_E with $L_{pre} = 100$ runs and $\mu(\delta T_E) = 1.384\,°$ and $\sigma_{RMSE}(\delta T_E) = 0.064\,°$.

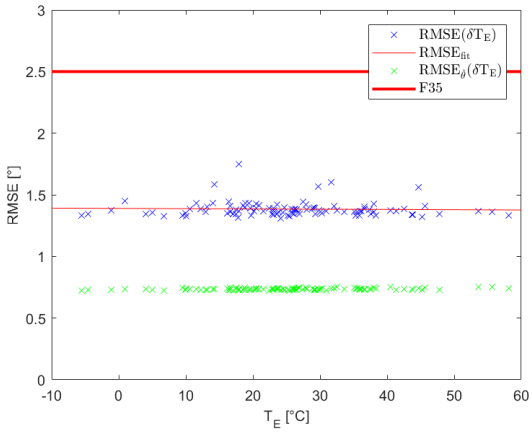


Fig. E-21: AKF scatter plot preliminary test result for T_E with $L_{pre} = 100$ runs.

E.4 Switching Time

For the actuator switching time a simulation related problem occurred which may possibly have a negative impact on the performance. The zero-crossing detection was deactivated for some simulation runs, resulting two levels of estimation performance on the P2PMAF+CF configuration Fig. E-22. The AKF was not affected by this issue.

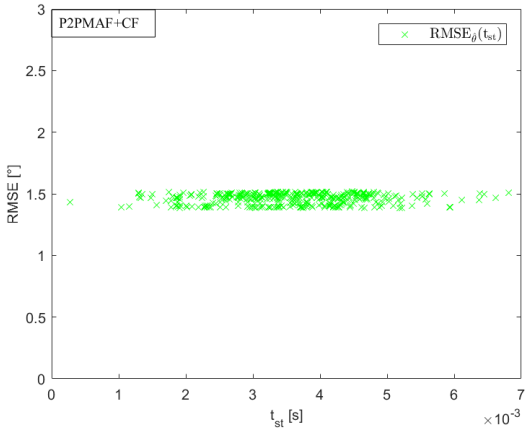


Fig. E-22: $RMSE_{\hat{\theta}}(t_{st})$ scatter plot for $L = 400$ runs.

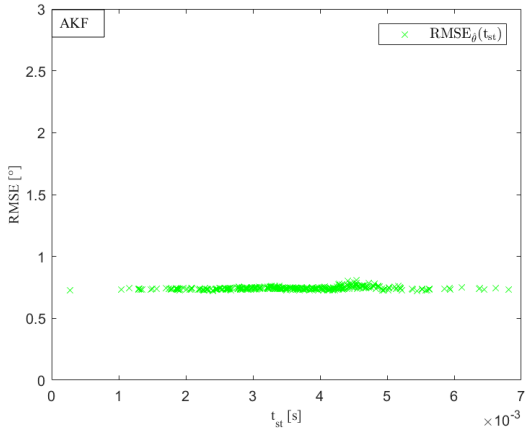


Fig. E-23: $RMSE_{\hat{\theta}}(t_{st})$ scatter plot for $L = 400$ runs.

E.5 Weight

To thoroughly explore the design space the range was increased to $R_{m_0} = 5\,\%$. Therefore, formulating the MATLAB input:

```
delta_m_0 = normrnd(5,5*0.05,400,1)
```

The graphs below show the histogram and the scatter plot of the P2PMAF+CF configuration.

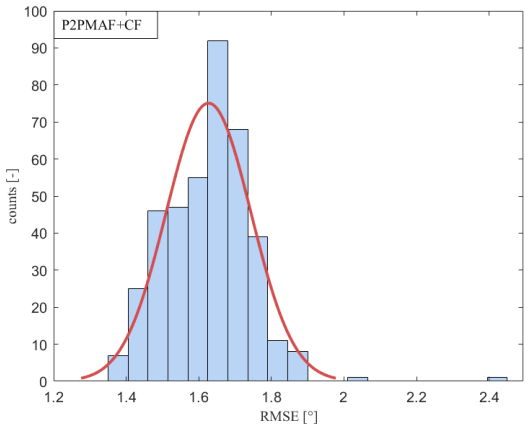


Fig. E-24: $RMSE$ histogram for m_0, $R = 5$ %, $\mu_{RMSE} = 1.627$ ° and $\sigma_{RMSE} = 0.117$ °.

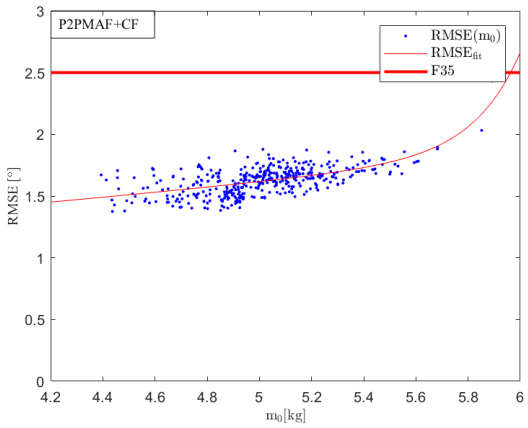


Fig. E-25: $RMSE$ scatter plot for m_0, $R = 5$ % and a second-order exponential fit.

The histogram indicates that the performance of the CF configuration has a probability of 68.27 % to fall within one standard deviation, $\sigma_{RMSE} = 0.117$ °, from the mean for the given input range.

$$p(RMSE = 1.627 \pm 0.117\ [°]) = 68.27\ [\%]. \qquad \textbf{Eq. (E–20)}$$

A second-order exponential model was fitted to the scatter plot. It can be approximated with the function:

$$RMSE_{fit}(m_0) = 0.828\, e^{0.1334 m_0} + 1.576 * 10^{-15} e^{5.647 m_0}\ [°] \qquad \textbf{Eq. (E–21)}$$

A maximum controllable weight is revealed and can be calculated with

$$m_{0,max,CF} = m_0\left(RMSE_{fit,CF} = F35 = 2.5\ [°]\right) = 5.963\ [\text{kg}]. \qquad \textbf{Eq. (E–22)}$$

The histogram with the PDF for the AKF configuration is shown in Fig. E-26 and the scatter plot in Fig. E-27.

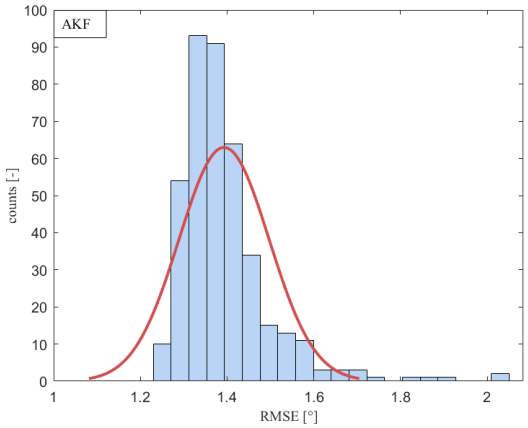


Fig. E-26: $RMSE$ histogram for m_0, $R = 5\,\%$, $\mu_{RMSE} = 1.393\,°$ and $\sigma_{RMSE} = 0.104\,°$.

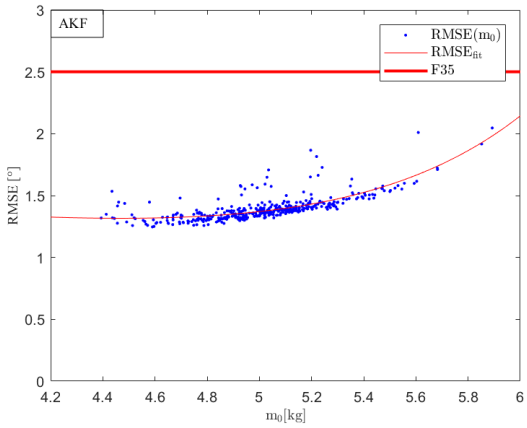


Fig. E-27: $RMSE$ scatter plot for m_0, $R = 5\,\%$ and second-order exponential fit .

The histogram indicates that the performance of the AKF configuration has a probability of $68.27\,\%$ to fall within one standard deviation, $\sigma_{RMSE} = 0.104\,°$, from the mean for the given input range.

$$p(RMSE = 1.393 \pm 0.104\,[°]) = 68.27\,[\%]. \qquad \textbf{Eq. (E–23)}$$

A second-order exponential model was fitted to the scatter plot. It can be approximated with the function:

$$RMSE_{fit}(m_0) = 2.338\,e^{-0.150 m_0} + 1.450 * 10^{-4} e^{1.503 m_0} \qquad \textbf{Eq. (E–24)}$$

A maximum controllable weight is revealed and can be calculated with

$$m_{0,max,AKF} = m_0\left(RMSE_{fit,AKF} = 2.5\,[°]\right) = 6.183\,[\mathrm{kg}]. \qquad \textbf{Eq. (E–25)}$$

E.6 Centre of Mass

The input distribution of the distance from the pivot point to the point mass location, l_{CM}, is calculated with a range of $R_{l_{CM}} = 5\,\%$. The value was chosen because the actual center of gravity might deviate by large quantities from the estimated reference value once the final RCS design is chosen. The results to the $R = 5\,\%$ range are comparable to the mass variation with the same range. The RCS fails to control the pendulum for large positive deviations from the reference value. This shall be considered for later design considerations. The MATLAB function with the corresponding values is:

```
delta_l_CM = normrnd(0.667,0.667*0.05,400,1);
```

The resulting plots for the CF+P2PMAF configuration are shown in Fig. E-28 and Fig. E-29.

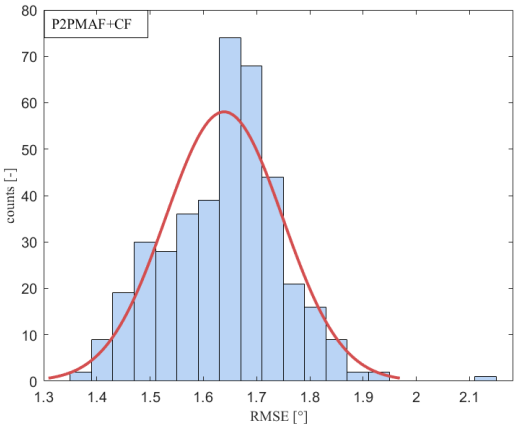


Fig. E-28: $RMSE$ histogram for l_{CM}, $\mu_{RMSE} = 1.634\,°$ and $\sigma_{RMSE} = 0.110\,°$.

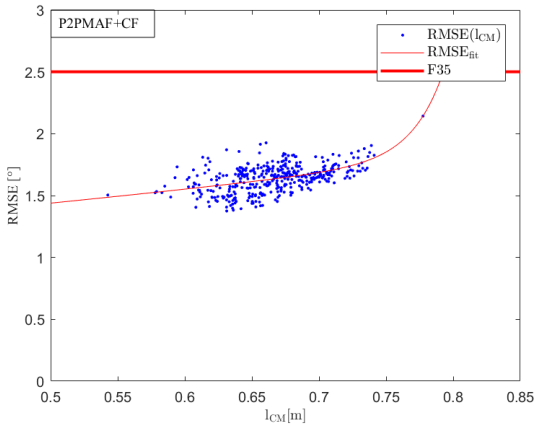


Fig. E-29: $RMSE$ scatter plot for l_{CM} and a second-order exponential fit.

The histogram indicates that the performance of the CF configuration has a probability of 68.27 % to fall within one standard deviation, $\sigma_{RMSE} = 0.110\,°$, from the mean for the given input range.

$$p(RMSE = 1.634 \pm 0.110\ [°]) = 68.27\ [\%]. \qquad \textbf{Eq. (E–26)}$$

A second-order exponential model was fitted to the scatter plot. It can be approximated with the function:

$$RMSE_{fit}(l_{CM}) = 0.986\, e^{-0.756 l_{CM}} + 1.382 * 10^{-15} e^{42.720 l_{CM}} \qquad \textbf{Eq. (E–27)}$$

The histogram with the PDF for the AKF configuration is shown in Fig. E-30 and the scatter plot in Fig. E-31.

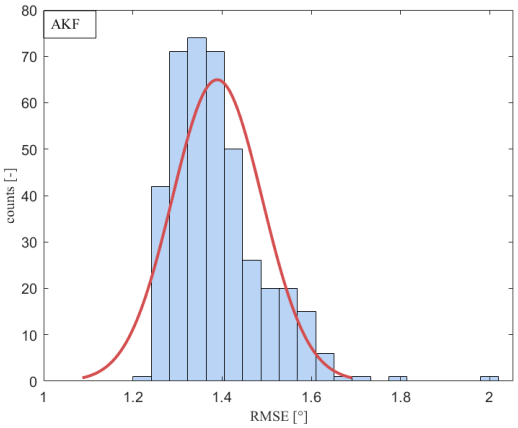


Fig. E-30: $RMSE$ histogram for l_{CM}, $R = 5\,\%$, $\mu_{RMSE} = 1.389\,°$ and $\sigma_{RMSE} = 0.101\,°$.

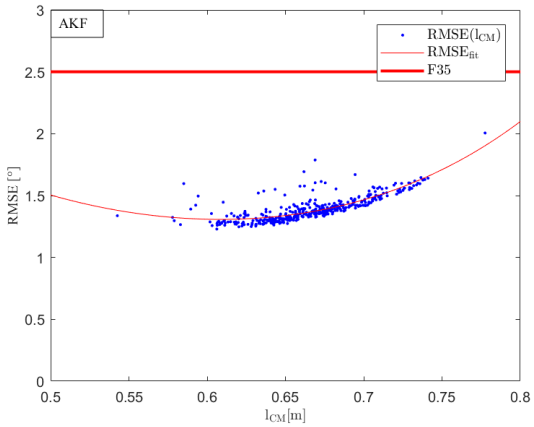


Fig. E-31: $RMSE$ scatter plot for l_{CM}, $R = 5\,\%$, and second-order exponential fit.

The histogram indicates that the performance of the AKF configuration has a probability of 68.27 % to fall within one standard deviation, $\sigma_{RMSE} = 0.060\,°$, from the mean for the given input range.

$$p(RMSE = 1.389 \pm 0.101\ [°]) = 68.27\ [\%]. \qquad \textbf{Eq. (E–28)}$$

A second-order exponential model was fitted to the scatter plot. It can be approximated with the function:

$$RMSE_{fit}(l_{CM}) = 10.540e^{-4.293 l_{CM}} + 0.012e^{6.226 l_{CM}}. \quad \textbf{Eq. (E–29)}$$

The output results for the two filter configurations are summarized in the table below.

Table E-1: Test results for both filter configurations with key factor l_{CM}.

Filter	Range	STD	Mean	PDS
P2PMAF+CF	5 [%]	0.110 [°]	1.634 [°]	0.558 [–]
AKF		0.101 [°]	1.389 [°]	1.067 [–]

E.7 Gyroscope Bias

The residual bias on the gyroscope can result from bias instability, described in 4.5 , or it can be the result of a poor calibration procedure. For consumer grade sensors there is also a ON / OFF bias change. The bias changes after deactivation and reactivation. The effect of the residual bias may lead to a drift in the estimation of the attitude. With the applied filters, this effect was successfully reduced. The following graphs show the influence of an equally distributed residual bias input over a range of $R_{\dot{\theta}_{RB,G,y}} = 50\ \%$.

P2PMAF+CF:

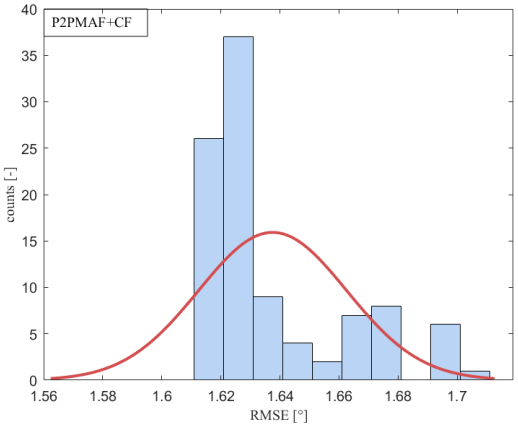


Fig. E-32: CF histogram preliminary test for $\dot{\theta}_{RB,G,y}$, $R_{\dot{\theta}_{RB,G,y}} = 50\ \%$ and $L_{pre} = 100$ runs.

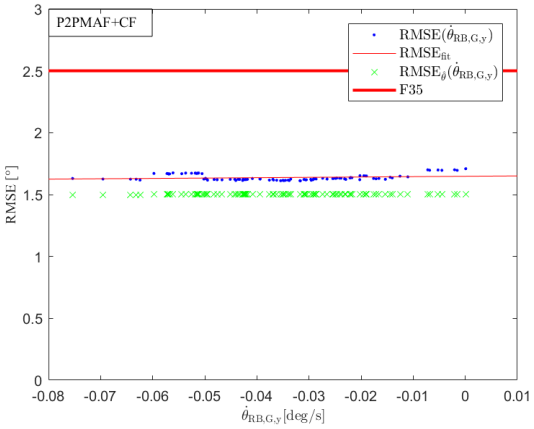


Fig. E-33: CF scatter plot preliminary test for $\dot{\theta}_{RB,G,y}$, $R_{\dot{\theta}_{RB,G,y}} = 50\ \%$ and $L_{pre} = 100$ runs and linear fit model.

AKF:

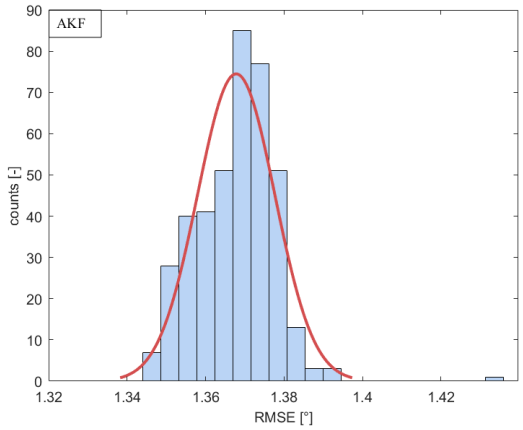


Fig. E-34: CF histogram preliminary test for $\dot{\theta}_{RB,G,y}$, $R_{\dot{\theta}_{RB,G,y}} = 50\,\%$ and $L = 400$ runs.

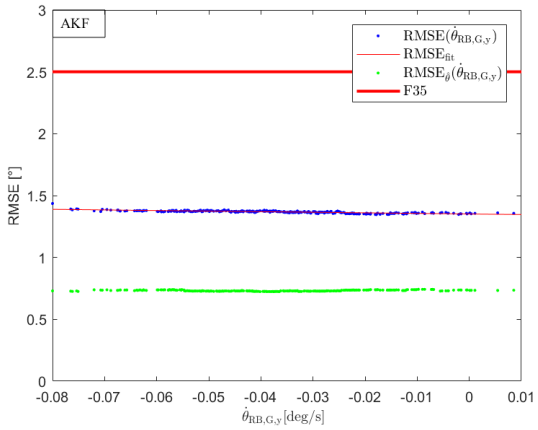


Fig. E-35: CF scatter plot preliminary test for $\dot{\theta}_{RB,G,y}$, $R_{\dot{\theta}_{RB,G,y}} = 50\,\%$ and $L = 400$ runs and linear fit model.

E.8 Accelerometer Noise

To formulate the intersection function between the fitted surface and F35, the accelerometer noise input was increased to:

```
delta_sigma_A = 0.1 + (0.5-0.1) * rand(400,2);
```

This resulted in the following scatter plots:

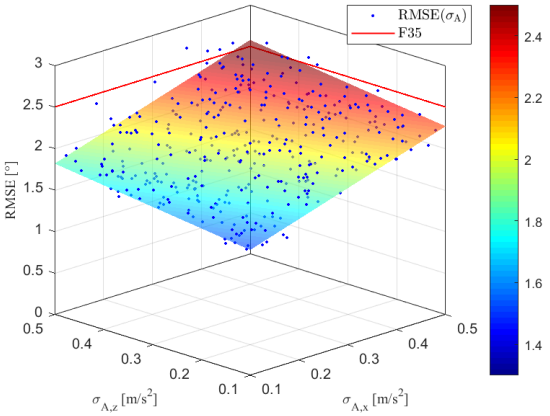


Fig. E-36: P2PMAF+CF $RMSE$ scatter plot for $\boldsymbol{\sigma}_A$ with multivariate linear regression fit.

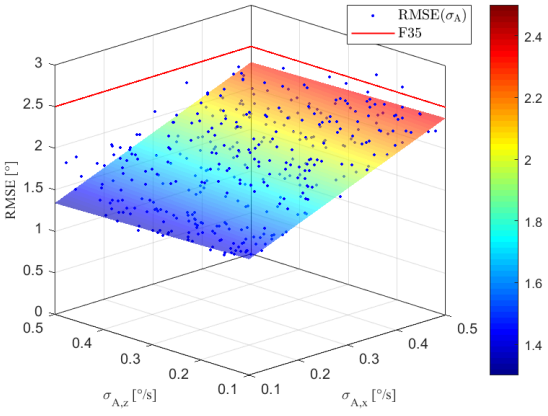


Fig. E-37: AKF $RMSE$ scatter plot for $\boldsymbol{\sigma}_A$ with multivariate linear regression fit.

E.9 Tuning Constant

The tuning constant C was reduced on a main test series setup to improve the result. It was reduced to $C = 0.5$, which succeeded in improving the performance of the AKF. The number of outliers was reduced to 7.

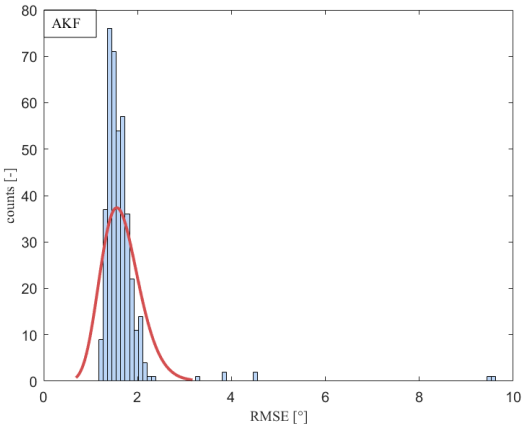


Fig. E-38: Response Gamma distribution for $C = 0.5$ with $\sigma_{RMSEtot} = 0.412\,°$ and $\mu_{tot} = 1.660\,°$.

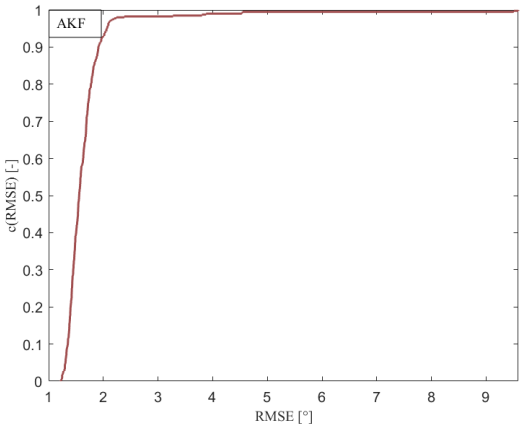


Fig. E-39: CDF with $c(\mu_{tot}) = 0.5$ and $C = 0.5$.

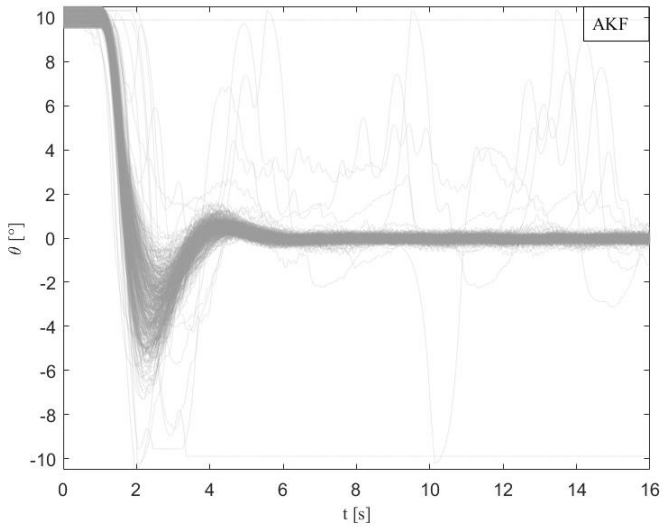


Fig. E-40: All 400 true trajectories for varying all key input factors with the given distributions and $C = 0.5$.