A New Quantitative Analysis for Prostheses Research and Evaluation

A Sensor Fusion Approach

J.W.G. Remmerswaal

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

A New Quantitative Analysis for Prostheses Research and Evaluation

A Sensor Fusion Approach

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

J.W.G. Remmerswaal

August 31, 2020

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Abstract

Problem : Due to high rejection rates regarding prostheses' use, the assessment of the amputee's use of the prosthesis has become more critical. Today's prosthesis research is limited to assessing a users' performance to perform tasks in a controlled environment. Therefore, these studies cannot wholly assess how the prosthesis is used in the daily lives of amputees. **Purpose** : The purpose of this thesis project is to create, test, and examine the performance of methods that can be used to enhance prostheses research and evaluation. **Results** : We have created several enhancements and extensions regarding prosthesis research and evaluation for three sensor scenarios. In the first scenario, we only use an accelerometer, which allowed us to create a new method for creating a vector magnitude (VM) that can show us what type of arm movement is made; additionally, we created a novel scoring system to determine the intensity of movements that are performed. In our second scenario, we used an additional gyroscope, which opened up possibilities for using more advanced Sensor Fusion (SF) techniques. We created an accurate tilt estimation algorithm that remains robust against high levels of gyroscope noise, accelerometer outliers, and measurement model violations. With the gyroscope, we were also able to create a VM from rotational velocity, which displays information about the arms' rotational movement. Additionally, we created a novel scoring system that also shows the intensity of the performed movements. Our last scenario considers the use of two Inertial Measurement Units (IMUs); we present a novel algorithm for estimating the relative sensor orientation. This algorithm uses a joint kinematic estimation method that incorporates the connection between adjacent segments within a SF algorithm that remains accurate in the vicinity of common real-world disturbances, among which are; high levels of gyroscope noise, accelerometer outliers, and Soft-Tissue-Artifacts (STA). **Conclusion** : The best way to enhance the analysis for prostheses research and evaluation is to start incorporating gyroscopes into the research process. This can either be in the form of the single sensor case or the double sensor system. The additional gyroscope(s) will enable the use the SF techniques and methods as discussed in this report.

Contents

List of Figures

List of Tables

Preface

When I still studied in Groningen, I once watched a YouTube video of a TEDx event, not knowing at the time that the inventor (dr. Gerwin Smit) of a lightweight prosthesis arm would eventually become my second supervisor during my thesis project. When I met with Manon for the first time, we quickly started talking about a combination of Sensor Fusion and something in the medical field. Luckily for me, Gerwin was interested in a collaboration to explore the possibilities of further enhancing prostheses' evaluation methods. After our first meetings, we started working on the project, and I could not be happier with the result.

Acknowledgements

First, I would like to thank my first supervisor, Dr. Manon Kok. During this whole year, you have helped me develop myself. I am very grateful that I have the pleasure of being your first graduate student at the TU Delft. I am also thankful for the chance you were willing to give me regarding the paper, which we will try to publish at the ECC 2021. Second, I would like to thank my second supervisor, Dr. Gerwin Smit. Thank you for your constructive attitude towards this project, it was really helpful and insightful for me to see and experience how two engineering fields can be combined. Still, it also showed me the struggles between different engineering languages when two areas are connected. This was an insightful experience from which I have learned a lot. I would also like to thank prof. dr. J.W. van Wingerden, when Manon asked my preference for the committee's senior member, it was you whom I named. The reason for this goes back to one of my first days at the TU Delft, where you helped me during one of the tutorials on bode plots and PID tuning. Thank you for being the senior member during my final colloquium. The last academic member I would like to thank is MSc. Ive Weygers. Ive thank you for sharing the data from the optical reference setup. I feel like, because of this, my thesis document has really increased in academic value, due to the comparisons I was able to make between your filter and mine.

I would also like to thank my fellow students' Rens and Matthijs, for a great first year. Thanks, Frida, Thomas, Daan, Bart, and Stefan, for our biweekly meetings, which were fun and insightful.

Bedankt mam, pap, Mick, Shane, Zinat zonder jullie had ik het niet gekund.

Delft, University of Technology — J.W.G. Remmerswaal
August 31, 2020

“Het begint met willen.”

— *Gerard Remmerswaal*

Glossary

List of Acronyms

IMU	Inertial Measurement Unit
IMUs	Inertial Measurement Units
SF	Sensor Fusion
MEKF	Multiplicative Extended Kalman Filter
ADLs	activities of daily living
BP	body powered
DH	Denavit-Hartenberg
DoF	Degrees of Freedom
jc	joint center
MEKF	Multiplicative Extended Kalman Filter
BM	bilateral magnitude
VM	vector magnitude
MR	magnitude ratio
RMSE	Root Means Square Error
SGH	Self Grasping Hand
AI	anatomically intact
PUs	prosthesis users'
PU	prosthesis user
SVD	Singular Value Decomposition
STA	Soft-Tissue-Artifacts
AC	activity counts

List of Symbols

δ	Dip angle
η	Orientation deviation state
ϕ	Roll
ψ	Yaw
θ	Pitch
κ	Learning Rate
ω	Angular velocity
Σ	Covariance
σ	Standard deviation
τ	Learning rate updater
b	Bias
e	Noise
G	Global frame
g	Gravity Vector
$G(s)$	Laplace transfer function
M	Marker frame
m	Magnetic field
n	Unit vector
O	Optical reference coordinate frame
q	Quaternion
R	Rotation matrix
r	Distance vector from joint center to Inertial Measurement Unit
S	Sensor frame
s	Laplace variable
T	Sampling time
t	Time index
w	Process noise
x	State vector
y	Measurement vector
$_{min}$	Minimum
$^\top$	Transpose
^{c}	Quaternion conjugate
^{L}	Left sided quaternion multiplication
^{R}	Right sided quaternion multiplication
$\mathbb{R}^{\mathrm{n}}$	Set of real numbers in n
$\mathcal{I}_{\mathrm{n}}$	n by n Identity matrix
det	Determinant

$\cdot$	Dot product
$\odot$	Quaternion multiplication
$\times$	Cross product

Chapter 1

Introduction

This introductory chapter defines the problem & motivation for this research, describing the intricacy in a problem-analysis and posing several research questions. After posing the research questions for this research, the main contributions of this work are highlighted. Finally, the introduction is concluded with the outline of this thesis document.

1-1 Problem-Statement and Motivation

Although there is a wide variety in available prostheses, rejection rates of upper limb prostheses are high [6]. The percentage of amputees that appear not to wear and use their device actively is estimated to be around 27%, and the percentage of people that ultimately reject the use of an artificial limb is 20% [53, 6]. These high rejection rates stem from different problems related to prostheses, e.g., pricing, life-like appearance, grip control, and weight [7]. Due to these high rejection rates, assessing the amputees' use of the prosthesis has become more critical. However, in today's world, prostheses research is limited to assessing a user's performance to perform tasks in a controlled environment and actively self-report questionnaires. These evaluations are obsolete and are not capable of thoroughly assessing how an amputee uses their artificial limb daily [16]. Due to the questions currently being raised on the methods for upper limb prostheses evaluation and the lack of complete assessment, other methods for proper evaluation of upper limb prosthesis activity are being explored. Body-worn motion sensors are now used more often as a non-invasive, objective, and accurate way to observe the movement of a subject [39, 56]. Recent studies of the upper artificial limbs are conducted using accelerometer data in the form of "activity counts" [2]. With these counts, the researchers create a track record of the prosthesis usage [16, 2]. Although making use of activity counts is a better way of assessing the use of a prosthetic limb in comparison to other methods, it is still limited. This is due to the activity counts being ill-defined and inaccessible for researchers that do not use patented software of a company named Actigraph. Furthermore, besides activity counts, more information can be extracted from accelerometer measurements that can be considered novel. This led to questions being raised on how to

enhance upper limb prostheses analyses further. In this thesis work, we will discuss several of these questions and explore the answers accordingly.

1-2 Problem Analysis

To create a thorough understanding of the problem that is being presented, we create a problem-analysis. Here, we first define the different components that make up the analyses of wearable systems like an artificial limb. The analyses and remote monitoring of wearable systems are said to consist of the following three parts [44].

- The communication hardware and software to relay data to a remote center.
- The sensing and data collection hardware to collect physiological and movement data.
- The data analysis techniques to extract clinically-relevant information from physiological and movement data.

Since the focus of this research project will mainly be on the improvement and enhancement of upper limb prosthesis analyses, the third point is the most significant concern. However, if we review the first and second bullet points, we can extract several constraints regarding this research and, thus, the problem. The first bullet point raises a question on the communication and relay of data. Therefore, it is vital to take into account the length of the data-set and know how it will be transferred from the sensor to a computer. In [16], a one-week data set is recorded with a sampling rate of 20 Hertz and stored within the sensor's memory; this data is then transferred to a computer after the participant is finished wearing the sensor. Recording data for one-week or longer leads to a final data-set, which is of an enormous magnitude. In this work, we will be aiming to find a solution that is capable of transferring and handling more or equal amounts of data, as presented in [16]. The second bullet point forces us to consider several options regarding the sensors, i.e., the hardware. Since the goal of this research is to enhance existing prostheses' evaluation methods, three options can be made regarding this; one can start by researching how to extract more information from state-of-the-art methods, i.e., accelerometer-only analyses. A second option could be to incorporate different sensors, when trying out different sensors, a starting point would be to use Inertial Measurement Units (IMUs) instead of accelerometers. The last option would be a combination of adding more different types of sensors, e.g., adding several IMUs on different parts of the body. The Inertial Measurement Unit (IMU) is a device that consists of an accelerometer and a gyroscope; the accelerometer measures a specific force that is on the body where the sensor is placed upon, while the gyroscope measures the angular velocity. Although inertial sensors are heavily used in other fields related to health and healthcare, e.g., walking analysis, improving posture stability, and fall detection, they have not yet been used very frequently in upper limb prostheses analysis [3, 60, 50]. The three options regarding the sensors to further improve upper artificial limb analysis are listed below in the following order.

- Extract more information within state-of-the-art methods that require an accelerometer.
- Add different sensors in the form of a gyroscope, which will translate to using an IMU instead of only using an accelerometer.

- Adding multiple sensors, whereby several IMUs are used to extract information for upper limb analyses.

This thesis will show how the three options for the improvement of prostheses analysis can be created. Most of these options require inertial Sensor Fusion (SF) techniques to arise; inertial SF is the umbrella term for fusion algorithms that combine measurements from different sensors to obtain new information. Since we cannot add an infinite amount of IMUs on the body, since this will not yield a practical application where we ask a participant to wear IMUs for at least seven days, we are left with a constraint regarding the number of sensors used. We evaluate two cases, a single sensor case, and a double sensor case. The single sensor case is defined as having one IMU per arm, in this case, around the wrist. The single sensor case can then be further distinguished into two parts: one where only an accelerometer is used and the case where an additional gyroscope is used. The double sensor case is defined as having two IMUs per arm, one on the wrist, and one on the upper arm. Most IMUs are used in combination with a magnetometer to obtain an absolute orientation estimate [23]. In almost all cases, a problem regarding the magnetometer may arise, disturbances caused by ferromagnetic materials can disturb the local magnetic field. When this occurs, the magnetometer will not be able to accurately measure the earth's magnetic field anymore [23]. Performing upper limb activity analysis of a prosthesis is one of these cases. Since it is most likely that the prosthesis will get into contact with a ferromagnetic material, no magnetometer can be used for this analysis. The prosthesis used for this research is the prosthesis presented in [54], i.e., the Self Grasping Hand (SGH), which can be seen in Figure 1-1 with and without a cosmetic glove. However, it should be noted that all methods presented in this work can be applied to other upper limb prostheses. For more information on prostheses in general and the SGH, please refer to Appendix B.

(a) The SGH with cosmetic glove , to cover up the mechanical appearance and give a more life-like look to an artificial limb [1].

(b) The SGH without cosmetic glove, where the button on the back of the hand is visible, used to open the SGH [36].

Figure 1-1: The SGH with and without cosmetic glove.

1-3 Objective

To ensure the relevance of this research project, the author has created various research questions in combination with a summary of the question's motivation.

Research Questions

- ***What is the best way to enhance the analysis of upper limb prostheses?***
  This is the key question that is the backbone of the research. To answer this question, we will need to develop different methods and strategies to extract new and relevant information from an amputee and evaluate these methods.
- ***How can we extract more information within state-of-the-art methods?***
  By looking at the accelerometer, we may be able to find new ways to analyze the prosthetic limb that have not been executed yet. One of the main objectives would be to bypass the "activity count" from [2].
- ***How can we enhance prostheses evaluation when we add different sensors?***
  By adding a gyroscope to the already existing solution of using an accelerometer, we enable ourselves to start using several SF methods to further enhance prostheses analyses.
- ***How can we enhance prostheses evaluation when we add more different type of sensors?***
  There exist methods that rely on the constraints imposed by the human body itself in the form of rigidly connected segments. Rigidly connected segments allow for accurate relative sensor orientation estimation when two or more IMUs are placed on adjacent segments [61].

1-4 Contribution

The goal of this research mainly consists of developing novel extensions to current upper limb prostheses analyses. These analyses are meant to be applied to humans who have lost their lower arm and who are undergoing testing and analysis regarding their prosthetic limb. The main contributions of this work are listed below.

Main Contributions

- ***Extension of an accelerometer-only analyses tool in the single sensor case***
  As an extension of the accelerometer-only approach, we have created a tilt estimation algorithm where the sensors are placed upon the wrist. We found that due to violations of the assumed measurement model, where we assumed that no external linear acceleration occurred, the inclination estimations became unreliable. Alongside our attempt to create the tilt estimation algorithm, we created a scoring system that does not rely on Actigraphs classified activity counts (AC) [2]. A new method for creating a vector magnitude (VM) is created as well as a novel scoring system, which introduces a rating to observe the intensity of the movements that are performed when only accelerometer measurements are available.
- ***Extension of an IMU analysis tool in the single sensor case***
  Extending the existing approach with a gyroscope opens up possibilities for using Algorithm 1 from [24]. Based on this, we created an extension that allows for robust tilt

estimation that does not require a magnetometer and is resistant to measurement model violations and accelerometer outliers. Additionally, an different activity scoring system was made, gyroscope measurements were used instead of accelerometer measurements.

- ***Extension of an IMU analyses tool in the double sensor case***
  Extending the approach from [24] with the joint kinematic model of [61], we developed an algorithm that can process large quantities of data computationally efficient. Simultaneously, the algorithm is proven to be robust against many disturbances, including; high gyroscopic sensor noise, accelerometer outliers, and low and moderate levels of Soft-Tissue-Artifacts (STA).

1-5 Thesis Outline

This document is structured as follows. Chapter 2 will shed light on the technical background knowledge required to understand how the extensions of the upper limb analysis tools are created. These include knowledge on the IMU, SF, the orientation parameterizations, but also information on the complementary filter, which is often used in this research.

Chapter 3, we derive our first extension tool. In this chapter, we show in great detail the steps taken to create a tilt estimation tool, which eventually is tested on simulation data. We also show how the first scoring system for the arms are created.

In Chapter 4, we show the possibilities by extending the types of sensors with an additional gyroscope. We show how a robust tilt estimator is created and how a complete movement scoring system can be made using an additional gyroscope.

In Chapter 5, the double sensor case is worked out. The derivations and techniques are described in great detail and are tested with state of the art data (optical reference system).

In Chapter 6, a discussion is given on this work, where we discuss all major finding in this work and the meaning behind those findings. We will explain how our finding relate to work conducted by other scientists and we will discuss the limitations of our findings.

Finally, Chapter 7 concludes all the work and findings that are presented in this document, and it summarizes the answers to the research questions that were previously posted. Also, some suggestions regarding future research in this direction are made where the author points out the main challenges and questions that are to be answered.

Chapter 2

Related Work

In this chapter, fundamental theoretical concepts at the basis of this work are extrapolated and adapted from the literature survey that was created prior to this thesis. In particular, state-of-the-art methods regarding prostheses analyses are presented as well as the principles of orientation estimation and Sensor Fusion (SF). Subsequently, the theory behind the algorithm from [24] for orientation estimation is explained. Finally, the theory behind the creation of a drift-free relative sensor orientation is discussed, which makes use of joint kinematic constraints. This chapter aims to create a basic understanding of the theory that will be presented in the chapters to come.

2-1 Prosthesis Analyses Using Accelerometers

As mentioned before, prostheses analyses are currently being conducted based on clinical tests of function and self-report questionnaires [16]. Lately, questions have been raised on the current state of affairs regarding the evaluation process. These questions are due to the lack of complete and well-rounded assessment. Other possible methods for evaluation are being explored. Accelerometers are frequently being used more as a way to observe movements by patients [39]. In [16] and [14], new methods for the analysis and display of upper limb activity monitoring data were presented by utilizing accelerometers. In [14], two new ways for the analysis and display of upper limb activity monitoring data are introduced and demonstrated with real-world data to show their potential value. In both studies, the GT3X+ Activity Monitors, from the company ActiGraph were used in combination with ActiGraph's patented algorithms within the Actilife software [46]. This software was used to filter and group data into one-second epochs, while subsequently converting it into activity counts (AC). Although the exact knowledge on the creation of the AC is kept as a company confidential secret, it is used in many studies see, e.g., [51, 57]. By counting an activity, it is meant that an activity that caused the acceleration signal to exceed a certain threshold is counted as an activity; anything below this threshold is ignored. At the end of the measurement period, the number of AC can be extracted from the software [2].

In [14], the prosthesis users' functionality is evaluated by assessing four different outcome measures. These four measures consist of the user being assessed for task completion, task duration, quality of movement, and gaze behavior. To evaluate the daily usage apart from the clinic, the symmetry in their real-world arm use is assessed using activity monitoring. The participants were asked to wear two accelerometers, one on each wrist (anatomical and prosthesis) and, to enable comparison with previously published data, the activity monitoring data were analyzed using the methods of Bailey [4].

Bailey's research [4] was conducted to quantify real-world upper-limb activity in adults that are non-disabled and adults with chronic stroke. A chronic stroke may lead to one of the upper limbs to become paretic, i.e., partially paralyzed. Simultaneously, the methods that were used there can also be applied to upper-limb prostheses research. In Bailey's research [4], participants were also asked to wear GT3X+ Activity Monitors on both wrists. After they were finished wearing the accelerometers, the data was filtered and grouped into one-second epochs and subsequently converted into AC. In each one-second epoch, AC across the three axes were added up, and the length of that vector was calculated as a vector magnitude (VM) as follows

$$\mathrm{VM} = \sqrt{\mathrm{AC}_x^2 + \mathrm{AC}_y^2 + \mathrm{AC}_z^2}. \tag{2-1}$$

This was done for both the paretic and non-paretic upper limb, with these two variables it is possible to derive a bilateral magnitude (BM) that represents the overall activity per second across the upper arms, which is created by summing up the VM of the paretic upper limb and the VM of the non-paretic hand as

$$\mathrm{BM} = \mathrm{VM}_{\mathrm{paretic}} + \mathrm{VM}_{\mathrm{Non-Paretic}} \tag{2-2}$$

From the paretic and non-paretic VM, it is also possible to derive a magnitude ratio (MR), which represents the relative contribution of each arm to the activity as

$$\mathrm{MR} = \ln\left(\frac{\mathrm{VM}_{\mathrm{Paretic}}}{\mathrm{VM}_{\mathrm{Non\ Paretic}}}\right) \tag{2-3}$$

This method does have some predicaments regarding unilateral activity, where the VM of one arm equals zero. This will then produce an infinite value for the MR, as can be seen in Figure 2-1. Hence, the arbitrary values of MR = 7 and MR = −7 are chosen as boundary values that represent unilateral use of the paretic and non-paretic limbs [4]. After this, the data is virtually represented by creating a scatter plot of the MR on the x-axis and the BM on the y-axis, which is supported by a color map that represents the number of occurrences in seconds of each point. Additionally, the median of the MR and BM are provided to report summary measures of the symmetricity and intensity of use for both limbs.

The methods created by Bailey provide good initial insight into the use of upper limbs. However, as stated in [16], the measure of contribution to an activity used in Bailey's study (MR) is based on the natural log and is therefore not intuitive; additionally, due to the arbitrary value introduced for unilateral activity, the scale of MR is not continuous. This

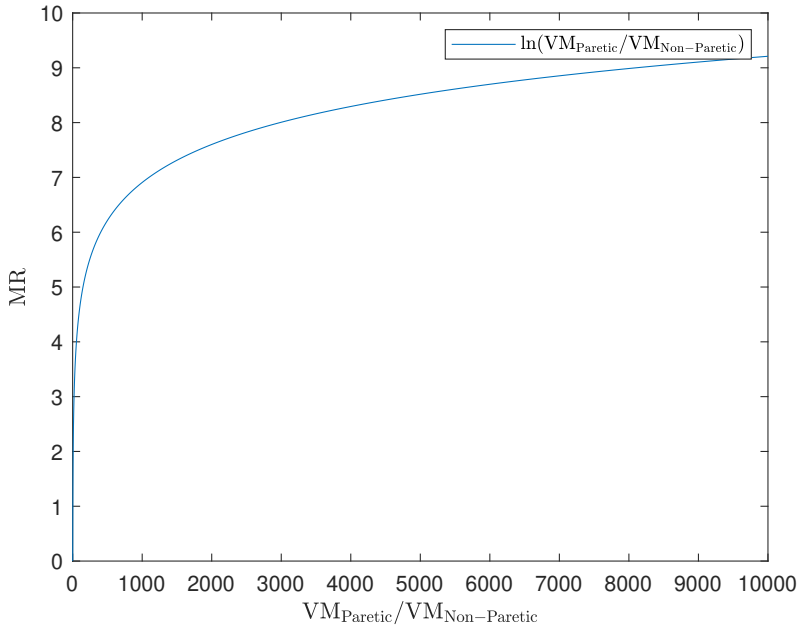


Figure 2-1: The problem regarding the MR from Bailey's method [4].

causes Bailey's method to have an abstract nature and, therefore, makes it hard to interpret. Hence, histograms that display a summary of the amount of activity across the upper limbs are proposed in [16]. By illustrating the data in the form of histograms, the researchers can visualize the contribution per arm to all the activities throughout the recording period. The contribution to activity between the upper limbs can be expressed as a ratio in percentages. In, [16] this percentage of contribution per arm per epoch was calculated by dividing the VM of the dominant/anatomical arm by the total VM across both arms, i.e., BM. In the case of the dominant/anatomical arm, the formula would then be

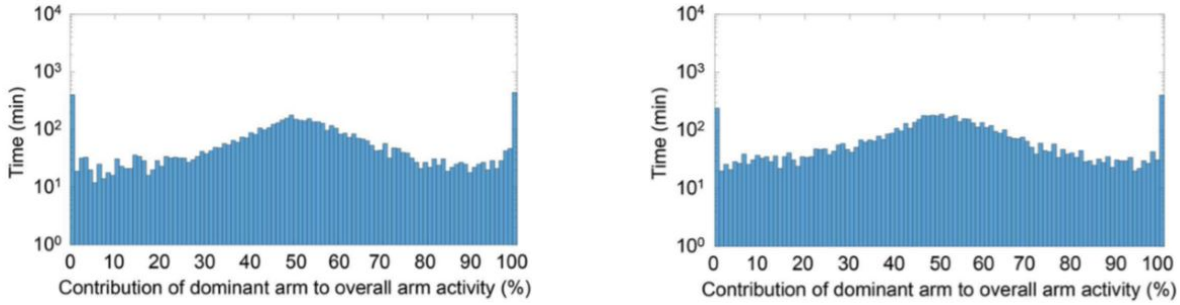
$$\text{Contribution}_{\text{Dom}} = \left[\text{round} \left(\frac{\text{VM}_{\text{Dom}}}{\text{VM}_{\text{Dom}} + \text{VM}_{\text{Non-Dom}}} \times 100 \right) \right], \tag{2-4}$$

and for the non-dominant

$$\text{Contribution}_{\text{Non-Dom}} = \left[\text{round} \left(\frac{\text{VM}_{\text{Non-Dom}}}{\text{VM}_{\text{Dom}} + \text{VM}_{\text{Non-Dom}}} \times 100 \right) \right]. \tag{2-5}$$

Whenever the VM of both arms was equal to zero, it was removed from the original data-set. After this, percentage bands from 0-100% with an increment of 1% were created, and the time in minutes was added up. This was then displayed on 100 histograms; simultaneously, a $\log_{10}$ scale was used for the indication of time. This was done for ease of visualization and interpretation. In Figure 2-2, the data recorded of two anatomically intact (AI) participants is shown. Here, the arms' contribution to the overall arm activity is mainly bilateral with unilateral activity bursts on both the dominant and non-dominant sides [16].

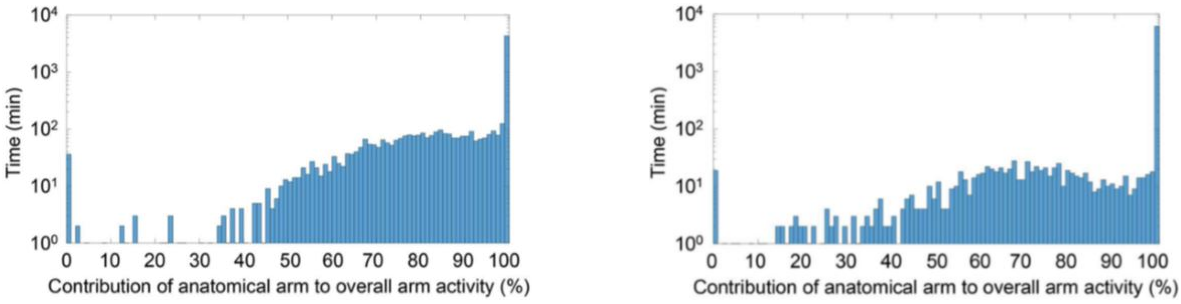
In Figure 2-3, the upper limb prosthesis users' (PUs) contribution towards the overall arm activity is depicted with the help of histograms. It can be seen that the PUs have a clear



(a) Right handed healthy AI 1 **(b)** Left handed healthy AI 2

Figure 2-2: The histogram plots from [16] of two AI adults.

preference for using their AI arm, and therefore it can be observed in the figures that the time contributed to overall arm activity is heavily skewed towards the AI arm [16].



(a) Trans-radial myoelectric prosthesis user (PU) 1 **(b)** Trans-radial myoelectric PU 2

Figure 2-3: The histogram plots from [16] of two adult PUs.

Additionally, to the problem of ease of interpretation, Bailey's method does not allow for the evaluation of temporal patterns [16]. Temporal patterns can be extremely relevant when evaluating prostheses since they allow the researcher to discover whenever the prostheses are used in multiple occurring situations [16]. To unravel these possible temporal use patterns, a new visualization tool for upper limb analyses was created. The Archimedean spiral plots in [16], firstly introduced as a visualization tool by [13], were created to visualize periodic serial data. The Archimedean spiral plots have been used in other research to visualize periodic human behavior, e.g., see [29]. Archimedean spirals have a straight line drawn from the origin that will intersect each spiral ring at the same time point in the data; this property allows for the revelation of possible temporal patterns. In [16] and [15], spiral plots were created where each epoch was marked with an event marker. If no external acceleration was measured, and thus no AC was created, i.e., BM = 0; the epoch was classified as 'both arms at rest.' If only one accelerometer measured an external force and thus only one arm created AC, this would be classified as 'unilateral' for the corresponding arm. When both arms performed a specific activity causing an external acceleration to be measured, this would then be classified as 'bilateral', where a percentage of contributions, as seen in the histograms, was calculated. These contribution percentages were split into 10% bands instead of the 1% that was used for the histogram. In Figure 2-4, the color scheme from [15, 16] is given, for every 10% band a different color was used, the colors were chosen in a complementary fashion to ensure that

the patterns of usage would be visible. The spirals were plotted as a 24-hour clock, where midnight would be at the top.

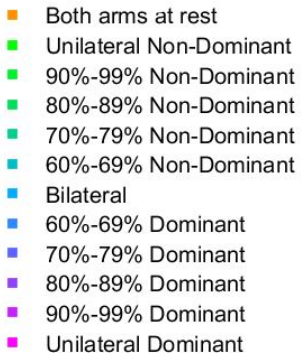


Figure 2-4: Color coding scheme of [15, 16].

With this color scheme, the Archimedean spiral plots were created with the same data used to create the histogram plots. In Figure 2-5, the data recorded from one of the two anatomically intact (AI) participants is shown. From this figure, it can be observed that the arms' contribution to the overall arm activity is mostly categorized as bilateral with sudden bursts of unilateral activity on both the dominant and non-dominant sides.

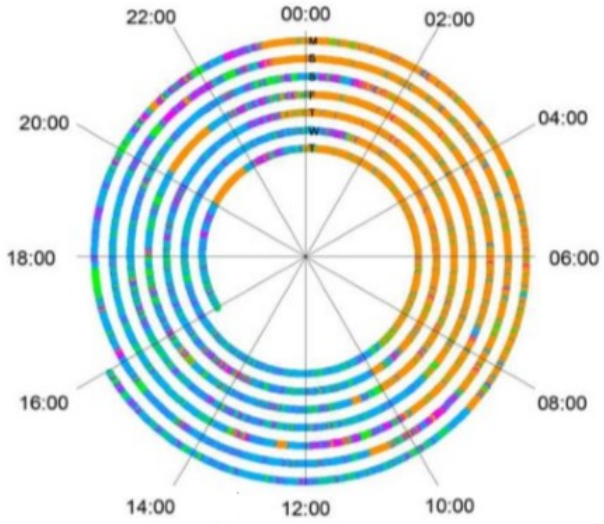


Figure 2-5: Archimedean spiral of an AI adult from [16], where mainly bilateral use of the arms is detected for AI adults.

The same figure can be created with the data recorded of one of the two upper limb PUs; this is shown in Figure 2-6. When a comparison is made between the PU and the AI human, it is observed that the PU shows little unilateral prosthesis activity and large amounts of single-use of the anatomical arm. Whenever the PU does perform bilaterally during an activity, there is a clear preference towards using the anatomical arm.

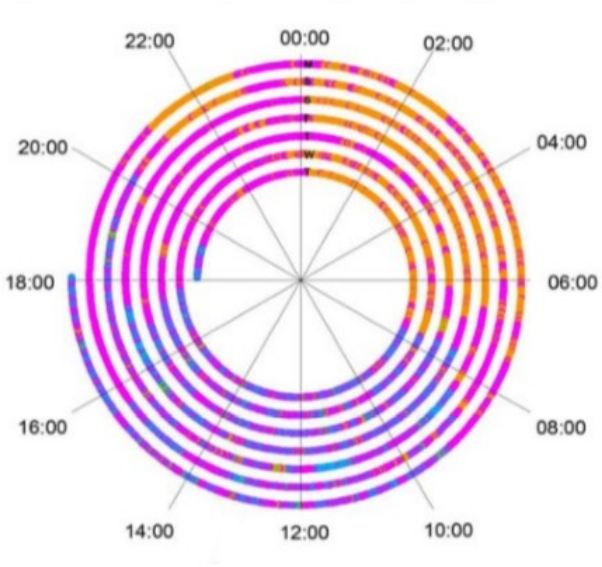


Figure 2-6: Archimedean spiral of an adult upperlimb PU from [16], where mainly unilateral use of the AI arm is seen.

To summarize, in [16], two new state-of-the-art visualization tools specifically directed towards the evaluation of upper limbs were created. First, histograms of AC, combined with simple descriptive statistics, were used to display the distribution of activity in between the two upper limbs. Secondly, Archimedean spiral plots are utilized to show how the participants use their upper limbs during the monitoring period. Using a color-coding scheme, it is possible to quickly observe how the person relies on his/her arm(s) or prosthesis. The combination of the two evaluation tools shows that it is possible to create a clear image of the differences in upper limb usage behavior between PUs and AI human. The most significant advantage of these plots over the methods that have been presented in [4] is that changes in the patterns of behavior can easily be identified. A suggestion given by the authors of [16], is the use of an under-laying thicker line around the existing spiral, to add additional information about the participant.

2-2 Orientation Estimation and Sensor Fusion

In this section, we present the theoretical foundation behind orientation estimation and SF. We will start by showing the motivation behind using SF with an example. After this, we explain different coordinate frames used throughout this research; they are explained together with measurement models that correspond to the different sensors and the dynamic models that can be used for orientation estimation. After this, we will explain a state-of-the-art complementary filter from [24].

2-2-1 Basic Orientation Estimation

As mentioned in the introduction, a gyroscope is a sensor that measures the angular velocity. One could argue that to create an estimate of the orientation, only the integration of the gyroscope measurement is required since the angular velocity is the derivative of the orientation; hence its integration would lead to a correct orientation estimate. A depiction of this process is given in Figure 2-7.

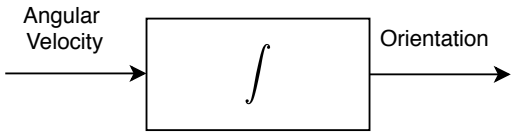
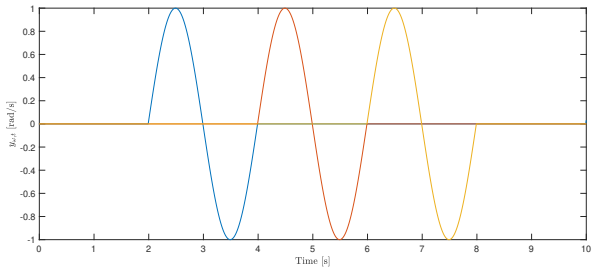


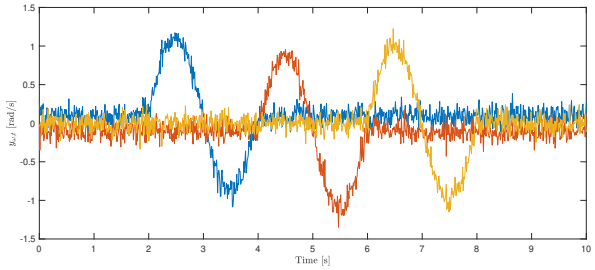
Figure 2-7: Block-scheme depicting the integration of the angular velocity, by which information on the orientation is obtained.

The scheme in Figure 2-7 would indeed be correct if the initial orientation would be known, and if a perfect measurement model would exist. Unfortunately, this is never the case because the gyroscope measurements contain some form of noise and bias. The bias and noise create a less accurate measurement; they also inhibit anyone from obtaining the correct absolute orientation by just plainly integrating the measured angular velocity. If one would directly integrate the gyroscope measurement, a phenomenon called integration drift will occur. This drift occurs in the orientation estimation due to the integration of the noise and bias [23]. To create an example to illustrate the effects of the bias and noise on the measured angular velocity, a time-varying angular velocity signal is created together with a noise and bias component. In this example the noise signal is distributed as $e_\omega(t) \sim \mathcal{N}(0, \sigma_\omega \mathcal{I}_3)$, with $\sigma_\omega = \frac{5\pi}{180}$ rad/s for all independent directions. The bias is assumed as a constant and is given as $\begin{bmatrix} 0.1 & -0.1 & 0 \end{bmatrix}^\top$ rad/s. In Figure 2-8a, a simulated measurement is taken when no bias nor noise occurs. In Figure 2-8b, a constant bias and the zero-mean Gaussian noise are added.

To obtain an estimate of the orientation that overcomes the issue of integration drift, a magnetometer is often used in combination with the inertial sensors, where the assumption is made that no other acceleration but the gravity is measured. The magnetometer can measure the local magnetic field. These drift-free estimates can be created by a fusion of the inertial sensor measurements with magnetometer measurements; this type of fusion is more often referred to as inertial SF [23]. For the most basic orientation estimation, inertial SF algorithms rely on the knowledge obtained from the gyroscope, the accelerometer, and the magnetometer. In a basic orientation estimation setup, the sensors only rotate and will not translate. The accelerometers will then only measure the local gravity vector since no linear acceleration is exerted on the sensor [23]. From this measured gravity vector, we can deduce information about the inclination of the sensor. As mentioned before, the magnetometer complements the other inertial sensor measurements. This is because it provides information about the heading of the sensor. In other words, it provides information about the orientation around the gravity vector, and this information cannot be deduced from accelerometer measurements only. The magnetometer can provide heading information all around the world except at the magnetic poles since the magnetic field is vertical [23]. An example is created to illustrate the workings of integration drift. In Figure 2-9 the errors of two orientation estimates of simulated data set are shown to illustrate the effects of integration drift. The data set contains a simulated



(a) Here the measured angular velocity is plotted against time, where we can observe that there is no noise nor bias component acting on the measured angular velocity.



(b) Here the same measured angular velocity is plotted against time, where we can observe that there is noise and a bias component acting on the measured angular velocity.

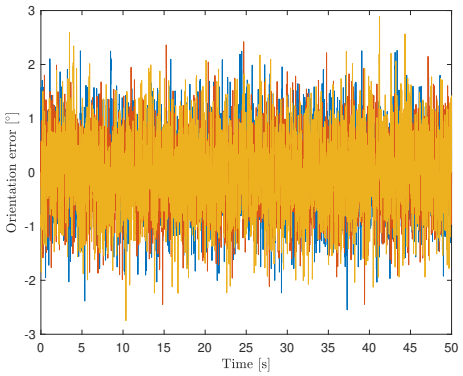
Figure 2-8: Measured angular velocity with no bias and no noise versus measured angular velocity with bias and noise.

Inertial Measurement Unit (IMU) that rotates around each axis. Subsequently, an additional noise component is added to the gyroscope, accelerometer, and magnetometer. In Figure 2-9a a near-zero error is observed. This is due to the SF algorithm for the orientation estimation using information that is given by the accelerometer and magnetometer. In Figure 2-9b, the orientation estimation only relies on the integration of the gyroscope measurements, as was depicted in Figure 2-7. Due to the integration of the noise component in the gyroscope measurement, drift in Figure 2-9b is observed.

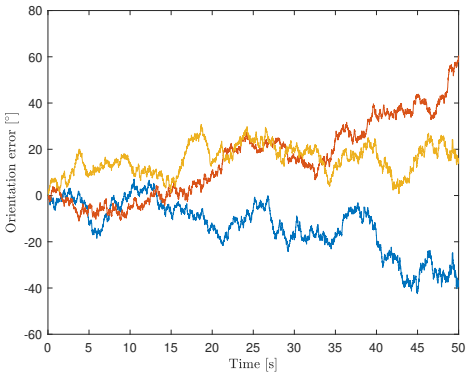
2-2-2 Coordinate Frames & Measurement Models

As mentioned above, fundamental orientation estimation with the help of SF techniques requires three different sensors; the gyroscope, the accelerometer, and the magnetometer. When discussing the different quantities measured by the inertial sensors, two coordinate frames need to be introduced:

- **The sensor frame** S is the coordinate frame of the IMU; the origin lies at the center of the accelerometer triad, which is aligned with its casing [23].
- **The global coordinate reference frame or global frame** G is a local geographic frame in which we want to navigate [23]. In most orientation estimation problems, we are interested in the S frame orientation with respect to the global frame G.



(a) Error in orientation when no integration drift occurs, x-axis (blue), the y-axis (red) and the z-axis (yellow).



(b) Error in orientation when integration drift occurs, x-axis (blue), the y-axis (red) and the z-axis (yellow).

Figure 2-9: No integration drift versus integration drift.

In the sensor frame S, it is possible to measure the angular velocity. However, as mentioned before, the gyroscope cannot generate a perfect measurement; this imperfection stems from the existence of a slowly time varying-bias and noise [23]. By assuming that the earth's rotation is negligible and the global frame G is static, the gyroscope measurement at time t can be modeled as

$$y_{\omega,t} = \omega_t^S + e_{\omega,t}^S. \tag{2-6}$$

Here, ω_t^S, represents the angular velocity, and $e_{\omega,t}^S$ the noise, which is typically assumed to be $e_{\omega,t}^S \sim \mathcal{N}(0, \Sigma_\omega)$. Here the superscript S represents that all these variables are in the sensor frame. We observed in Figure 2-8b, that the gyroscope measurement can contain a bias. If this is the case the measurement model of Eq. (2-6) can be extended with an extra variable to represent the bias, $b_{\omega,t}^S$. The bias is a time-dependent variable since it can vary over time; the gyroscope bias can change slowly due to different effects such as temperature change or exposure to impact [33]. Modeling the measurement in this fashion is a popular choice and can be found in several research papers; e.g., see [28, 40, 61]. When properly calibrated, the measurements of the three gyroscope axes will be independent of one another. This implies that the covariance can be assumed as

$$\Sigma_\omega = \begin{bmatrix} \sigma_{\omega,x}^2 & 0 & 0 \\ 0 & \sigma_{\omega,y}^2 & 0 \\ 0 & 0 & \sigma_{\omega,z}^2 \end{bmatrix}. \tag{2-7}$$

The slowly time-varying bias $b_{\omega,t}$ can be treated in different ways. If the experiment is short, it is possible to pre-calibrate or estimate the bias as a constant. If not, it can be assumed to be slowly time-varying, and the bias can be chosen as a part of the state vector [23]. Another option is to ignore the bias when it is not the focus of research and has negligible effects [34].

The other sensor placed within an IMU is the accelerometer. The accelerometer measures a specific force that is on the body where the sensor is placed. By neglecting the centrifugal and

acceleration and the Coriolis acceleration of the earth, and assuming the linear acceleration to be approximately zero [23], the accelerometer measurements can be modeled as

$$y_{a,t} = -R(q_t^{SG})g^G + e_{a,t}^S. \tag{2-8}$$

Here g^G represents the gravity vector, that is being rotated from the global frame G, to the sensor frame S, by the rotation matrix $R(q_t^{SG})$. Here the superscript SG represents the rotation from the global frame to the sensor frame, which is aligned with the sensor axes. Just like the gyroscope measurements, the accelerometer measurements includes a time-varying bias $b_{a,t}^S$ and a noise term, which can be modeled as $e_{a,t}^S \sim \mathcal{N}(0, \Sigma_a)$. It should be noted that this model is seldom completely true, since linear external accretion may occur, caused by movement. Nevertheless, it can be assumed that the model is a decent approximation since external linear acceleration for humans is generally a lot lower than the gravity force. It should also be noted that not in all literature, the measurement is modeled in this manner, in some literature; e.g., see [61, 40], the linear acceleration is assumed to be non-zero. If the measurement model is chosen in that manner, the measurement model $y_{a,t}$ is as follows

$$y_{a,t} = R(q_t^{SG})\left(a^G - g^G\right) + e_{a,t}^S. \tag{2-9}$$

Regarding the time-varying bias, $b_{a,t}^S$, the same options hold as seen when discussing the gyroscope bias. It should be noted that for self-calibration, a completely flat surface is required. This has to do with the gravity vector being rotated and thus also partially divided among the x- and y-axis when it is put on a non-flat surface. When someone then tries to measure the bias, they will end up removing a part of the gravity vector that is supposed to be there.

The magnetometer measures the local magnetic field m^G and is used to complement the inertial sensor measurements. This local magnetic field has its horizontal component pointing towards the earth's magnetic north pole. The ratio between the horizontal and vertical components, which depends on the position on the earth is referred to as the dip angle δ.

As mentioned before, the magnetometer complements the other inertial sensor measurements. This is because it provides information about the heading of the sensor. It provides information about the orientation around the gravity vector, and this information cannot be deduced from accelerometer measurements only. The magnetometer can provide heading information all around the world except at the magnetic poles since the magnetic field m^G is vertical [23]. The magnetic field vector is modeled into components as follows

$$m^G = (\cos\delta \quad 0 \quad \sin\delta)^\top. \tag{2-10}$$

Here, it is assumed without loss of generality that $\left\|m^G\right\|_2 = 1$, by assuming that the magnetometer will only measure the local magnetic field, the measurements can be modeled as

$$y_{m,t} = R(q_t^{SG})m^G + e_{m,t}^S, \tag{2-11}$$

Where the noise term is assumed to be $e_{m,t}^S \sim \mathcal{N}(0, \Sigma_m)$.

2-2-3 Dynamic Models & Initial Orientation Estimates

Sensor orientations are dynamic and therefore have an initial orientation and a corresponding dynamical model that describes the change in orientation over time. Orientations can be described in different orientation parameterizations; there are rotation matrices, Euler angles, axis-angle representations, and unit quaternions. For a detailed description and information on these orientation parameterizations and their derivations, please refer to Appendix C. To describe the discrete-time dynamic models, it is first required to create an overview of the continuous-time model for the angular velocity. The derivatives of the orientation, e.g., the angular velocity can be written as

$$\frac{dq^{GS}}{dt} = q^{GS} \odot \frac{1}{2}\omega_q^S. \tag{2-12}$$

Here q^{GS} is a unit-quaternion, which represents a rotation from the sensor frame, S with respect to the global frame, G and ω_q^S is a representation for the angular velocity expressed as a quaternion and the $\odot$ represent the quaternion multiplication which is

$$p \odot q = \begin{bmatrix} p_0 q_0 - p_v \cdot q_v \\ p_0 q_v + q_0 p_v + p_v \times q_v \end{bmatrix} = p^L q = q^R p. \tag{2-13}$$

The superscripts L and R denote the left side multiplication and right side multiplication respectively and are denoted as

$$p^{\mathrm{L}} \triangleq \begin{bmatrix} p_0 & -p_v^\top \\ p_v & p_0 \mathcal{I}_3 + [p_v \times] \end{bmatrix}, \quad q^{\mathrm{R}} \triangleq \begin{bmatrix} q_0 & -q_v^\top \\ q_v & q_0 \mathcal{I}_3 - [q_v \times] \end{bmatrix}. \tag{2-14}$$

The cross product $[u\times]$ is defined as

$$[u\times] \triangleq \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}. \tag{2-15}$$

For a derivation of this relation, please refer to [20]. By utilizing an Euler discretization and assuming that there is a constant angular velocity between two time-samples, the dynamics of the orientation can be expressed in terms of the sampling time T and the angular velocity. This way of modeling can be seen in large amounts of research literature, e.g., see [24, 34, 61].

$$q_t^{GS} = q_{t-1}^{GS} \odot \exp_{\mathrm{q}}\left(\frac{T}{2}\omega_t^S\right) \tag{2-16}$$

Where the $\exp_{\mathrm{q}}$ is the following operator as

$$\exp_{\mathrm{q}}(x) = \begin{bmatrix} \cos \|x\|_2 \\ \frac{x}{\|x\|_2} \sin \|x\|_2 \end{bmatrix}. \tag{2-17}$$

In the problem of estimating orientation by using inertial sensors, there are two standard modeling options. The first option consists of a state vector x_t containing the orientation and angular velocity.

$$x_t = \begin{bmatrix} (q_t^{GS})^\top & (\omega_t^S)^\top \end{bmatrix}^\top \tag{2-18}$$

For the dynamic model of the angular velocity, it is possible to use motion models. Examples of such models are the constant angular velocity model. These models contain a process noise model, $w_{\omega,t}^S \sim \mathcal{N}(0, \Sigma_\omega)$. This noise term is there to model the assumptions on how constant the angular velocity is. The motion model is given as

$$\omega_t^S = \omega_t^S + w_{\omega,t-1}^S. \tag{2-19}$$

Another possibility for describing the state vector is only to include the orientation, i.e., $x_t = q_t^{GS}$. Here the gyroscope measurements can then be used as an input to the dynamic equation $x_{t+1} = f_t(x_t, y_{\omega,t}, w_t)$. Thus the change in the orientation is directly modeled in terms of the inertial measurements. This means that the angular velocity, as seen in Eq. (2-16), will be obtained from the measurements generated by the gyroscope. The advantage of using a motion model for the state dynamics is that knowledge about the motion of the sensor can be incorporated in this model. However, it does create the disadvantage of having a large state vector. The advantage of using the inertial measurements as inputs to the dynamics is that the process noise can be represented as the inertial measurement noise. Also, the changes in angular velocity will have a faster effect on the state. This is since the constant motion model delays the effect of changes in the dynamics. Hence, those estimates tend to look slightly more smooth [23].

To run an orientation estimation algorithm, an estimate of the initial orientation is required. In the current literature, several methods are found to determine the initial orientation. Some of these methods are relatively accurate, and some are crude guesses. The first method to determine the initial orientation, is the TRIAD algorithm [23, 34]. This method utilizes the first magnetometer and accelerometer measurements. Unless the magnetometer measurements are obtained at the magnetic north or south poles, the first measurement will contain information about the heading. This method uses the property that with two or more linearly independent vectors in two coordinate frames, it is possible to determine the rotation between the two coordinate frames [23]. With the TRIAD method, four vectors will be obtained, the accelerometer measurement $y_{a,t}$, the magnetometer measurement $y_{m,t}$, the local gravity vector g^G and the local magnetic field vector m^G. The accelerometer measurements contain information about the sensor's inclination, whereas the magnetometer measurements contain information about the heading. However, it should be noted that the magnetometer also contains information about the inclination; this is due to its non-zero vertical component. In real-world applications, it is typically the case that the accelerometer gives more accurate information about the inclination [23]. Hence the magnetometer is only used to obtain information about the heading by projecting the magnetic field vector and the magnetometer measurement on the horizontal plane. If we now normalize the vectors, the following is obtained.

$$\begin{aligned} \hat{g}^G &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^\top, \quad \hat{g}^S = \frac{y_{a,1}}{\|y_{a,1}\|_2}, \\ \hat{m}^G &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^\top, \quad \hat{m}^S = \hat{g}^S \times \left(\frac{y_{m,1}}{\|y_{m,l}\|_2} \times \hat{g}^S \right). \end{aligned} \tag{2-20}$$

The initial orientation can be obtained by formulating an optimization problem [23, 34], where the orientation q^{GS} minimizes the distance between the normalized magnetic field and the gravity vector that is measured at the first time step and the normalized magnetic field and gravity vector in the global frame. This optimization problem is then defined as

$$\begin{aligned} \underset{q^{GS}}{\arg\min} \quad & \left\| \bar{\hat{g}}^G - q^{GS} \odot \bar{\hat{g}}^S \odot q^{SG} \right\|_2^2 + \left\| \bar{\hat{m}}^G - q^{GS} \odot \bar{\hat{m}}^S \odot q^{SG} \right\|_2^2, \\ \text{subj. to} \quad & \left\| q^{GS} \right\|_2 = 1. \end{aligned} \tag{2-21}$$

If we now define

$$A = -\left(\bar{\hat{g}}^G\right)^L \left(\bar{\hat{g}}^S\right)^R - \left(\bar{\hat{m}}^G\right)^L \left(\bar{\hat{m}}^S\right)^R, \tag{2-22}$$

where the superscripts ^{L} and ^{R} stand for the left and right quaternion multiplication as defined in Eq. (2-14). The optimization problem can also be written as

$$\begin{aligned} \hat{q}_1^{GS} = \arg\min_{q^{GS}} \quad & \left(q^{GS}\right)^\top A q^{GS}, \\ \text{subj. to} \quad & \left\| q^{GS} \right\|_2 = 1. \end{aligned} \tag{2-23}$$

The solution for this problem is equal to the eigenvector corresponding to the largest eigenvalue of the matrix A [10]. As mentioned before, the TRIAD algorithm uses a magnetometer, just like the methods used in [40, 22]. In [40], the initial heading of the sensors was found by only using magnetometer measurements from one segment. In [22], all variables were initialized at zero except for the orientations at the first time step, which are initialized around their estimated orientation. This estimated absolute orientation is also obtained by utilizing a magnetometer. Next to this method, there are other methods to perform orientation initialization without magnetometer information, e.g., see [61, 52, 41]. In [61], the initial orientation is set to $\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^\top$, in [52] it was demonstrated that, regardless of the choice of initial estimates, convergence to the true values was obtained both in simulation and experiment. This was proven by generating random values for both the initial inclination and heading, and in [41], one random initial estimate of the inclination and heading is generated for each data segment.

2-2-4 Complementary Filter

When performing orientation estimation, a popular choice is to use estimation filters. Many estimation filters have been created, where each filter has its own unique set of properties. It was already mentioned that the gyroscope and the combination between an accelerometer

and magnetometer contain information about the orientation [23, 24]. A complementary filter uses the fact that an orientation estimate generated by the accelerometer and magnetometer is noisy, but accurate over long periods. While orientation estimates generated by the integration of the gyroscope measurements yield accurate orientation estimates on a short time scale but will drift heavily over longer time horizons [23, 24]. For the orientation estimation of long data-sets, the complementary filter from [24] is an excellent option. This is because the complementary filter from [24] has the property that it is relatively computationally cheap. After all, no optimization is used. Another benefit of this filter is that it has an outstanding built-in mechanism to be very robust against outliers. A common disturbance comes in the form of the fact that human body segments are non-rigid. The sensors are placed on soft tissue, which can lead to the violation of the assumption that the sensor position with respect to the joint is fixed at all times. These types of disturbances are known as Soft-Tissue-Artifacts (STA) [27]. Another form of outliers that should be taken into account is accelerometer outliers. In the real world, these can be caused by the sensor or the body that the sensor is placed upon banging or ticking against a hard surface; this will then cause a spike in the accelerometer measurement. Based on the measurement models and dynamical models shown in Section 2-2-2 and 2-2-3, the complementary filter from [24] is presented below. However, it should be noted that in [24], it is that both the accelerometer and magnetometer measurements are normalized before using them in the measurement models presented in section 2-2-2. The orientation estimate from the gyroscope measurements is denoted as q_{ω}^{GS} and the orientation estimate from the accelerometer measurements as q_{am}^{GS}. Then a complementary filter will put a low-pass filter on q_{am}^{GS} and a high-pass filter on q_{ω}^{GS}. This can be written as

$$\hat{Q}^{GS}(s) = G(s)Q_{am}^{GS}(s) + (1 - G(s))Q_{\omega}^{GS}(s), \tag{2-24}$$

here s represents the Laplace variable and $G(s)$ denotes the transfer function of the filter, which is given by $G(s) = \frac{1}{as+1}$. $Q^{GS}(s)$ stands for the orientation q^{GS}, when it is transformed in the Laplace domain and $\hat{q}^{GS}$ is the resulting filtered orientation [24]. When Eq. (2-24) is brought back to the discrete time-domain via a backward Euler

$$\hat{q}_t^{GS} = (1 - \gamma_t)\, q_{am,t}^{GS} + \gamma_t \left(\hat{q}_{t-1}^{GS} + T\omega_{q,t}\right), \tag{2-25}$$

is obtained. Here T denotes the sampling time, $\gamma_t = \frac{a}{a+T}$, and $\omega_{q,t}$ represents the angular velocity expressed in terms of a quaternion. This conversion to express the angular velocity as a quaternion is given as

$$\omega_{q,t} = \frac{1}{2} S\left(\hat{q}_{t-1}^{GS}\right) y_{\omega,t}, \tag{2-26}$$

where

$$S(q) = \begin{bmatrix} -q_v \\ q_0 \mathcal{I}_3 - [q_v \times] \end{bmatrix}, \tag{2-27}$$

here, q_0 and q_v are elements of the unit quaternion as presented in Section C-4. The orientation estimation from accelerometer and magnetometer measurements can be formulated as an optimization problem where

$$\begin{aligned}\min_{\eta_t} V(\eta_t) = \min_{\eta_t} \frac{1}{2} \left\| y_{a,t} + \left(\exp_{\mathrm{R}}(\eta_t)\right)^{\top} R\left(\tilde{q}_t^{SG}\right) g^G \right\|_2^2 + \\ \frac{1}{2} \left\| y_{m,t} - \left(\exp_{\mathrm{R}}(\eta_t)\right)^{\top} R\left(\tilde{q}_t^{SG}\right) m^G \right\|_2^2\end{aligned} \tag{2-28}$$

In the objective function, use is made of the acceleration measurement model, when no linear acceleration is assumed. The orientation is written in terms of a linearization point and an associated deviation as

$$R\left(q_t^{GS}\right) = R\left(\tilde{q}_t^{GS}\right) \exp_{\mathrm{R}}(\eta_t), \tag{2-29}$$

and

$$\begin{aligned}\exp_{\mathrm{R}}(\eta_t) &= \mathcal{I}_3 + \sin\alpha[v\times] + (1-\cos\alpha)[v\times]^2, \\ &\approx \mathcal{I}_3 + [\eta_t\times].\end{aligned} \tag{2-30}$$

Rewriting the problem in this manner allows for optimization over an orientation deviation parameterized in terms of a rotation vector. For more information and a more in-depth explanation of encoding the orientation in terms of a linearization point that is parameterized as orientation deviation, please refer to Appendix C-5. By doing this, issues regarding quaternion normalization are avoided, and the optimization variable is reduced from a four-state variable to a three-state variable.

Instead of solving the objective function for each time step, a single gradient descent iteration is performed in [24]. This drastically decreases the computational time required to solve the problem, and due to the high sampling rates of the IMU, the corrections that need to be made are typically not large, and the estimates will converge over time. Linearising $V(\eta_t)$ around $\tilde{q}_t^{SG} = \hat{q}_{t-1}^{SG}$ and $\eta_t = 0$, the gradient descent step is given by

$$\hat{\eta}_t = -\mu_t \nabla V(\eta_t), \tag{2-31}$$

with

$$\begin{aligned}\nabla V(\eta_t) = -\left[R\left(\hat{q}_{t-1}^{SG}\right) g^G\times\right]\left(y_{a,t} + R\left(\hat{q}_{t-1}^{SG}\right) g^G\right) + \\ \left[R\left(\hat{q}_{t-1}^{SG}\right) m^G\times\right]\left(y_{m,t} - R\left(\hat{q}_{t-1}^{SG}\right) m^G\right),\end{aligned} \tag{2-32}$$

where μ_t is the gradient descent step length. The estimate of the orientation deviations state can now be used to compute $q_{am,t}^{GS}$

$$q_{am,t}^{GS} = \hat{q}_{t-1}^{GS} \odot \exp_{\mathrm{q}}\left(\frac{1}{2}\hat{\eta}_t\right) \approx \hat{q}_{t-1}^{GS} + \frac{1}{2} S\left(\hat{q}_{t-1}^{GS}\right) \hat{\eta}_t, \tag{2-33}$$

If Eq. (2-26) is substituted into Eq. (2-25), then the time update is obtained as

$$\hat{q}_t^{GS} = \hat{q}_{t-1}^{GS} + \frac{1}{2} S\left(\hat{q}_{t-1}^{GS}\right)\left(\gamma_t T y_{\omega,t} - \mu_t\left(1-\gamma_t\right)\nabla V(\eta_t)\right). \tag{2-34}$$

What is left, is choosing the values for γ_t and μ_t. When the estimation mainly relies on the integration of the gyroscope measurement and uses the accelerometer measurements to correct for integration drift one typically chooses $\gamma_t \approx 1$. In [24], the scaling factor for the

Algorithm 1 Orientation estimation using the complementary filter with orientation deviation states from [24].

Input: Gyroscope & normalized accelerometer measurement $y_{\omega,t}y_{a,t}$, orientation estimate from previous time instance q_{t-1}^{GS}, tuning parameter β and sampling time T.
Output: Orientation estimate $\hat{q}_t^{GS}$.

1: Compute

$$\nabla V(\eta_t) = -\left[R\left(\hat{q}_{t-1}^{SG}\right) g^G\times\right]\left(y_{a,t} + R\left(\hat{q}_{t-1}^{SG}\right) g^G\right) + \left[R\left(\hat{q}_{t-1}^{SG}\right) m^G\times\right]\left(y_{m,t} - R\left(\hat{q}_{t-1}^{SG}\right) m^G\right)$$

with $y_{a,t}$ and $\hat{q}_{t-1}^{GS}$

2: Based on $\nabla V(\eta_t)$ compute

$$\hat{\omega}_t = y_{\omega,t} - \beta\frac{\nabla V(\eta_t)}{\|\nabla V(\eta_t)\|}$$

3: Obtain the updated orientation

$$\hat{q}_t^{GS} \approx \hat{q}_{t-1}^{GS} + \frac{T}{2} S\left(\hat{q}_{t-1}^{GS}\right)\hat{\omega}_t,$$

with $S\left(\hat{q}_{t-1}^{GS}\right)$ as defined in Eq. (2-27).

gradient descent direction, $\mu_t(1-\gamma_t)$, is chosen to be $\frac{\beta T}{\|\nabla V(\eta_t)\|}$. This is done to enhance the algorithm's robustness against measurement model imperfections. The choice of β depends on the amount of drift that is to be expected from the gyroscope measurement integration. When the noise is assumed to be $e_{\omega,t} \sim \mathcal{N}(0, \sigma_\omega^2)$. Integration of the gyroscope measurements in one dimension results in an integration drift distributed as $Te_{\omega,t} \sim \mathcal{N}(0, T^2\sigma_\omega^2)$ [24]. it is known that the gyroscope measurement is a three-dimensional vector, the standard deviation of the integration drift for the unit quaternion is then given by $\sqrt{3}\sigma_\omega T$ [24]. This is a good choice for βT. The resulting filter equations are

$$\hat{q}_t^{GS} \approx \hat{q}_{t-1}^{GS} + \frac{T}{2} S\left(\hat{q}_{t-1}^{GS}\right)\hat{\omega}_t, \tag{2-35}$$

with

$$\hat{\omega}_t = y_{\omega,t} - \beta\frac{\nabla V(\eta_t)}{\|\nabla V(\eta_t)\|}. \tag{2-36}$$

The final algorithm for the complementary filter using orientation deviation states is given in Algorithm 1.

2-2-5 Bias Estimation

The gyroscope measurement model used to create an estimate of the orientation in Section 2-2-4 can be optionally extended with a gyroscope bias. This was explained and described in (2-6) in Section 2-2-2. A tool that can be incorporated to increase the accuracy of the

estimation algorithm is a gyroscope bias estimation algorithm. By estimating the offset of the gyroscope, it can be subtracted to obtain a better-uncontaminated measurement. In [24], the bias estimation method of [31] is used. In [31], the bias estimate is created by using the normalized direction of the estimated error in the rate of change of the orientation, which is denoted as $\hat{\dot{q}}_e^{GS}$. It is possible to express this as the angular error in each gyroscope axis as

$$\omega_{e,t}^S = 2\left(\hat{q}_{t-1}^{GS}\right)^c \odot \hat{\dot{q}}_{e,t}^{GS} \tag{2-37}$$

Here, Eq. (2-37) can be derived as the inverse to the relationship defined in equation Eq. (2-12). The gyroscope bias, $b_{\omega,t}^S$ is represented by the DC component of $\omega_{e,t}^S$ and can be removed as the integral of $\omega_{e,t}^S$, which is weighted by an appropriate gain ζ as

$$b_{\omega,t}^S = \zeta \sum_t^S \omega_{e,t}^S T \tag{2-38}$$

After this, a compensated gyroscope measurement, $y_{\omega_c,t}^S$ can be obtained as

$$y_{\omega_c,t}^S = y_{\omega,t}^S - b_{\omega,t}^S \tag{2-39}$$

2-3 Relative Sensor Orientation Estimation

In most cases, a problem regarding the magnetometer may arise, disturbances caused by ferromagnetic materials can disturb the local magnetic field. When this occurs, the magnetometer will not be able to measure the earth's magnetic field anymore [23]. There exist estimation methods that are closely related to the body and its kinematics, which require no magnetometer but do require the use of more than one IMU. These methods rely on the constraints imposed by the human body itself in the form of rigidly connected segments. Rigidly connected segments allow for accurate relative sensor orientation estimation when two or more sensors are placed on adjacent segments [61]. With information about these constraints and connected segments, it is possible to retrieve information about the joint axis position and joint angles. Here we will discuss the background knowledge required to obtain a relative sensor orientation estimate and how these estimates can be validated. We start by discussing the differences in biomechanical modeling, followed by the mathematics behind body constrained estimation; finally, we will explain how an optical reference system can be used to conclude whether the estimation algorithm works.

2-3-1 Biomechanical Models

A critical problem to consider is how the inertial sensor measurements can be related to the human body. It is possible to look at the human body as a system of separate links connected through joints. These joints allow the links to move around each other in a constrained manner [23]. These human body models are referred to as biomechanical models. This section will discuss the two most common biomechanical models, i.e., the kinematic chain model and the free segments model [34]. In the kinematic chain model, the global orientation, the position of

the root segment, and the relative orientation between the different segments are modeled as estimation variables [34]. The kinematic chain model is generally parameterized minimally, e.g., Euler angles or Denavit-Hartenberg (DH) coordinates. For more information on DH coordinates please refer to Appendix C-6. The modeling via a kinematic chain brings several advantages [34]. Kinematic chain models can be used to predict the body acceleration at the position of the IMU, which helps in the separation from accelerations due to gravitational forces. Another advantage is that restricted joint Degrees of Freedom (DoF) can easily be modeled by omitting single angular DoF. However, it should be noted that there are two restrictions on this advantage; the rotation parameterization needs to be in minimal form, and the coordinate systems should be sufficiently well aligned with the anatomical rotation axes. Lastly, the estimated variables (joint angles) in the model are simultaneously the object of interest in many applications [34, 35]. Unfortunately, there are also several disadvantages to using a kinematic chain model. The parameterization used for the model is mostly chosen to be minimal; these minimal orientation parameterizations suffer from singularities. Another disadvantage is that the calibration parameters, e.g., the lengths of the different segments and the poses of the sensors relative to the model, are assumed to be fixed and known. The lengths of the body segments are user-specific parameters that need to be determined only once, but the sensor poses and, in particular, their orientation relative to the segments must be re-calibrated every time the sensor network is placed on the subject's body [9, 8]. The free segment model chooses a non-minimal parameterization for the biomechanical model by representing each segment with an orientation and position with respect to a global reference frame. The orientation is mostly represented in singularity-free unit quaternions [9]. Conditions imposed by the biomechanical model, e.g., rigidly connected segments and restricted rotational DoF, can be included in the estimation via stochastic constraints.

2-3-2 Constrained Estimation

Two connected body segments of a human can be modeled as a system consisting of two adjacent rigid segments connected via a spherical joint, which have no restrictions regarding their rotational DoF [23, 61]. Both segments contain an inertial sensor with an individual sensor coordinate frame denoted by S_i, where $i = 1, 2$. The two segments are connected via the joint center (jc). The position vector $r_i^{S_i}$, defines the distance from the jc to the origin of the sensor coordinate frame [52]. To compensate for drift in the relative sensor orientation $R(q_t^{S_1S_2})$, use is made of the accelerometer and gyroscope measurements to update both orientations $R(q_t^{GS_1})$ and $R(q_t^{GS_2})$. Since the jc is a common point of both segments, it can be deduced that this point should have a unique acceleration $a_{jc,t}^{S_i}$ in a common reference coordinate frame. This can be expressed in the following equation

$$R(q_t^{GS_1})a_{jc,t}^{S_1} = R(q_t^{GS_2})a_{jc,t}^{S_2} + e_{link,t}. \tag{2-40}$$

Here it is assumed that the residual vector $e_{link,t} \sim \mathcal{N}(0, \Sigma_{link})$. The jc accelerations can be approximated by utilizing the acceleration measurements at a distance r_i from the jc by using $\mathcal{C}_t^{S_i}$ [61, 52, 28, 40, 41].

$$\begin{aligned} a_{jc,t}^{S_i} &= y_{a,t}^{S_i} - \mathcal{C}_t^{S_i} r_i, \\ \mathcal{C}_t^{S_i} &= \left[y_{\omega,t}^{S_i}\times\right]^2 + \left[\dot{y}_{\omega,t}^{S_i}\times\right]. \end{aligned} \tag{2-41}$$

Here $\dot{y}_{\omega,t}^{S_i}$ represents the time derivative of the angular velocity. For a detailed description on how Eq. (2-41) is obtained, please refer to [19]. It should be noted that this model only works properly as long as there is an acceleration next to the earth's gravity. Hence the measurement model in Eq. (2-8) should be assumed. In [61], the distance vector from the jc to the sensors is calculated according to the method presented in [52], which also makes use of Eq. (2-40) and Eq. (2-41). Here an iterative optimization Gauss-Newton algorithm is used to find the parameters $r = [r_1^\top \; r_2^\top]^\top$, which minimizes a sum of squared errors cost function as follows

$$\widehat{r} = \arg\min_{r} \sum_{t=1}^{N} e_{t,r}^2. \tag{2-42}$$

Here the error corresponds to the length of the error generated when the acceleration in the jc is calculated from the two sensor coordinate frames.

$$e_{r,t} = \left\| y_{a,t}^{S_1} - \mathcal{C}_t^{S_1} r_1 \right\|_2 - \left\| y_{a,t}^{S_2} - \mathcal{C}_t^{S_2} r_2 \right\|_2, \tag{2-43}$$

with its derivatives with respect to r as

$$\begin{aligned} \frac{\partial e_{r,t}}{\partial r_1} &= -\frac{{\mathcal{C}_t^{S_1}}^\top e_{1,r_1,t}}{\|e_{1,r_1,t}\|_2}, \\ \frac{\partial e_{r,t}}{\partial r_2} &= -\frac{{\mathcal{C}_t^{S_2}}^\top e_{2,r_2,t}}{\|e_{2,r_2,t}\|_2}, \end{aligned} \tag{2-44}$$

where

$$e_{i,r_i,t} = y_{a,t}^{S_i} - \mathcal{C}_t^{S_i} r_i, \quad i = 1, 2. \tag{2-45}$$

Another method that can be used to estimate the distance from the jc to the sensor is presented in [40]. Here the author shows how to obtain the distance from the jc to the sensor by minimizing the cost function of Eq. (2-43) as a sum of absolutes instead of a sum of the squared errors as

$$\widehat{r} = \arg\min_{r} \sum_{t=1}^{N} \left| e_{t,r} \right|. \tag{2-46}$$

A well-known property of this type of cost function is robustness against outliers in the data; this is because large residuals are weighted by their absolute value and not by their squared value in the cost function. To solve the optimization problem of Eq. (2-46), it is not possible to use a Gauss-Newton method since it is not a sum of squared errors minimization problem, other gradient descent type methods are required to solve this problem [11], which require the partial derivatives as

$$\begin{aligned}\frac{\partial e_{r,t}}{\partial r_1} &= -\operatorname{sgn}(e(k,r))\frac{\mathcal{C}_t^{S_1 T} e_{1,r_1,t}}{\|e_{1,r_1,t}\|_2},\\ \frac{\partial e_{r,t}}{\partial r_2} &= \operatorname{sgn}(e(k,r))\frac{\mathcal{C}_t^{S_2 \top} e_{2,r_2,t}}{\|e_{2,r_2,t}\|_2},\end{aligned} \tag{2-47}$$

with $e_{i,r_i,t}$ as Eq. (2-45) and

$$\operatorname{sgn}(x) = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases} \tag{2-48}$$

In [61], two estimation algorithms are presented; a filtering estimate, and a optimization based algorithm. For the filtering estimate the Multiplicative Extended Kalman Filter (MEKF) from [23] is used, which requires the encoding of the quaternion into a orientation deviation state as $\eta_t = \begin{bmatrix}\eta_{\mathrm{S}_1,t}^\top, & \eta_{\mathrm{S}_2,t}^\top\end{bmatrix}^\top \in \mathbb{R}^6$. After the estimation of the orientations from the first and second sensor frame to the global frame the relative sensor orientation angles can be obtained by the following equation

$$\hat{q}_t^{S_1 S_2} = \left(\hat{q}_t^{GS_1}\right)^c \odot \hat{q}_t^{GS_2}. \tag{2-49}$$

2-3-3 Estimation Validation

One of the methods for comparing the estimate to the actual orientation is via an optical reference system. Utilizing an optical reference system is widely used and accepted as a method to validate someone's findings [61, 28, 22]. After testing, a coordinate frame alignment is required since the sensor coordinate frame, and the marker-based coordinate frame is not perfectly aligned. Furthermore, the sensors' estimates are estimated with respect to a different coordinate frame than the orientation obtained via the optical reference system. In most cases, the sensor estimate will be estimated with respect to a global reference frame, and the information from the optical reference frame obtains an orientation with respect to its reference coordinate frame, where each marker also has a marker frame. To validate the estimation algorithm via an optical reference system, we introduce two additional coordinate frames next to the ones introduced in Section 2-2-2.

- **The optical marker frame** M is the coordinate frame of the markers that are being tracked by an optical reference system.
- **The optical reference system base coordinate frame** O is also a local geographic frame in which we want to navigate when utilizing optical reference tracking.

The alignment can be done via theorem 4.2 (Relative orientation from orientation measurements) from [20]. Here it is stated that, if we suppose that $\left\{q_t^{GS}\right\}_{t=1}^N$ and $\left\{q_t^{OM}\right\}_{t=1}^N$ satisfy $q_t^{GS} \odot q^{SM} \approx q^{GO} \odot q_t^{OM}$. Then the objective function can be formulated as

$$V\left(q^{GO}, q^{SM}\right) = \sum_{t=1}^{N} \|e_t\|_2^2 = \sum_{t=1}^{N} \left\| q_t^{GS} \odot q^{SM} \odot q_t^{OM} \odot q^{OG} - 1 \right\|_2^2. \tag{2-50}$$

Then Eq. (2-50), is minimized by $\hat{q}^{GO} = v_1$ and $\hat{q}^{SM} = u_1$, where u_1 and v_1 are the first left and right singular vectors from the Singular Value Decomposition (SVD) of matrix A, which is

$$A = \sum_{t=1}^{N} \left[\left(q_t^{GS} \right)^L \right]^\top \left[\left(q_t^{OM} \right)^R \right]. \tag{2-51}$$

Chapter 3

Accelerometer-only analysis of prosthesis usage

In the introduction, we mentioned that there are three extensions regarding the sensors. In this chapter, we discuss the first analysis extensions that can be created from the state-of-the-art method; hence we will only look at the case when only one accelerometer per arm is used. We will show how Sensor Fusion (SF) may aid in the analyses of the upper limb, but we will also show and discuss limitations regarding the created extensions. This chapter will introduce two extension ideas and show how they are worked out mathematically. The two extension on the state-of-the-art method can be split up into two parts: the extension to create an estimate of the inclination (roll (ϕ) & pitch (θ)), and the extension to bypass the dependencies on companies that provide build-in software tools to perform research, e.g., Actilife software from Actigraph. After the creation of the first extension, we will test the ideas on simulated data sets created in MATLAB. Next to this, we will use measurements from Xsens Inertial Measurement Units (IMUs) to create experimental results. For the second extension we will use a data-set, which is created by wearing one Axivity AX6 on each wrist for one day.

3-1 Tilt Estimation with an Accelerometer

Tilt estimation can be performed when one has access to an accelerometer. This is because the accelerometer measures the gravity vector. As mentioned in Section 2-2-1, from this measured gravity vector, we can deduce information about the inclination of the sensor. In this section, we discuss the mathematical concept behind tilt estimation from an accelerometer. Below we present the derivation of a quaternion that posses information about the inclination of the sensor [58]. Because of the Earth's gravity depends on the sensor's location and since the unit in which the acceleration can be expressed may vary, we assume that

$$\begin{aligned} y_{a,t} &= \begin{bmatrix} y_{a_x,t} & y_{a_y,t} & y_{a_z,t} \end{bmatrix}^\top, \quad \|y_{a,t}\|_2 = 1 \quad \forall\, t, \\ g^G &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^\top. \end{aligned} \tag{3-1}$$

With this notation, we can describe the measurement of the accelerometer in its most pure form. This means that for this problem, we equal the accelerometer measurement with the rotating gravity vector in the sensor frame; additionally, no bias and no noise are taken into account.

$$y_{a,t} = -R(q_t^{SG})g^G \tag{3-2}$$

If we would work out the equation with the rotation matrix when described in quaternions, we obtain the following equation

$$\begin{aligned} \begin{bmatrix} y_{a_x,t} \\ y_{a_y,t} \\ y_{a_z,t} \end{bmatrix} &= - \begin{bmatrix} 2q_0^2 + 2q_1^2 - 1 & 2q_1q_2 - 2q_0q_3 & 2q_1q_3 + 2q_0q_2 \\ 2q_1q_2 + 2q_0q_3 & 2q_0^2 + 2q_2^2 - 1 & 2q_2q_3 - 2q_0q_1 \\ 2q_1q_3 - 2q_0q_2 & 2q_2q_3 + 2q_0q_1 & 2q_0^2 + 2q_3^2 - 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \\ &= \begin{bmatrix} 2q_0q_2 - 2q_1q_3 \\ 2q_0q_1 - 2q_2q_3 \\ q_1^2 + q_2^2 - q_0^2 - q_3^2 \end{bmatrix}. \end{aligned} \tag{3-3}$$

If we expanded the system into three parts, we would obtain

$$\begin{cases} y_{a_x,t} &= 2q_0q_2 - 2q_1q_3, \\ y_{a_y,t} &= 2q_0q_1 - 2q_2q_3, \\ y_{a_z,t} &= q_1^2 + q_2^2 - q_0^2 - q_3^2. \end{cases} \tag{3-4}$$

The system above is underdetermined and thus has an infinite number of possible solutions. This is an expected result since the gravity vector's alignment from its global frame into the sensor frame does not give any information about the rotation in the heading direction. Rotations can achieve the gravity vector's alignment with a determination of the roll and pitch angles and an arbitrary yaw angle. To create a solvable system, we require the restriction of one variable. Therefore, we set $q_0 = 0$, and the system becomes

$$y_{a_x,t} = -2q_1q_3, \tag{3-5a}$$
$$y_{a_y,t} = -2q_2q_3, \tag{3-5b}$$
$$y_{a_z,t} = q_1^2 + q_2^2 - q_3^2. \tag{3-5c}$$

The system above is fully determined; thus solvable. The first step is to convert Eq. (3-5a) and Eq. (3-5b) into Eq. (3-5c). This is done by rewriting q_1 and q_2 into the accelerometer signal as

$$q_1 = -\frac{y_{a_x,t}}{2q_3}, \quad q_2 = -\frac{y_{a_y,t}}{2q_3}. \tag{3-6}$$

Now we substitute Eq. (3-5a) and Eq. (3-5b) into Eq. (3-5c) in the following fashion to obtain

$$\begin{aligned} y_{a_z,t} &= \left(-\frac{y_{a_x,t}}{2q_3}\right)^2 \left(-\frac{y_{a_y,t}}{2q_3}\right)^2 - q_3^2, \\ 4y_{a_z,t}q_3^2 &= y_{a_x,t}^2 + y_{a_y,t}^2 - 4q_3^4, \\ q_3^4 - y_{a_z,t}q_3^2 &+ \frac{\left(y_{a_z,t}^2 - 1\right)}{4} = 0. \end{aligned} \tag{3-7}$$

By setting q_3^2 equal to Q we get the following solvable quadratic equation

$$\mathrm{Q}^2 - y_{a_z,t}\mathrm{Q} + \frac{\left(y_{a_z,t}^2 - 1\right)}{4} = 0. \tag{3-8}$$

By solving the system for Q, we get the following two solutions

$$\mathrm{Q}_{1,2} = \frac{y_{a_z,t} \pm 1}{2}. \tag{3-9}$$

Obviously we will choose the solution that always yields a real number, which is

$$q_3 = \pm\sqrt{\frac{y_{a_z,t} + 1}{2}}. \tag{3-10}$$

Now we can use the solution of q_3 to calculate q_1 and q_2 by substituting Eq. (3-10) into Eq. (3-6) to obtain

$$q_1 = \pm\frac{y_{a_x,t}}{\sqrt{2\left(y_{a_z,t}+1\right)}}, \quad q_2 = \pm\frac{y_{a_y,t}}{\sqrt{2\left(y_{a_z,t}+1\right)}}. \tag{3-11}$$

Both q_1 and q_2 yield two equivalent solutions; this is due to the unit quaternion property of switching signs between one solution and the other. For ease of use, we choose the solution that has a positive quaternion element of q_1. The unit-quaternion then becomes

$$\hat{q}_t^{SG} = \begin{bmatrix} 0 & \frac{y_{a_x,t}}{\sqrt{2(y_{a_z,t}+1)}} & \frac{y_{a_y,t}}{\sqrt{2(y_{a_z,t}+1)}} & \sqrt{\frac{y_{a_z,t}+1}{2}} \end{bmatrix}^\top \tag{3-12}$$

What should be noticed is that a singularity problem arises whenever $y_{a_z,t} = -1$. To overcome this problem, we need to create another quaternion that does not have the same singularity issue; this quaternion can be made by setting either q_1 or q_2 equal to zero. For $q_1 = 0$ this gives

$$\hat{q}_t^{SG} = \begin{bmatrix} \frac{y_{a_x,t}}{\sqrt{2(1-y_{a_z,t})}} & 0 & \sqrt{\frac{1-y_{a_z,t}}{2}} & \frac{y_{a_y,t}}{\sqrt{2(1-y_{a_z,t})}} \end{bmatrix}^\top, \tag{3-13}$$

and for $q_2 = 0$

$$\hat{q}_t^{SG} = \begin{bmatrix} -\frac{y_{a_y,t}}{\sqrt{2(1-y_{a_z,t})}} & \sqrt{\frac{1-y_{a_z,t}}{2}} & 0 & \frac{y_{a_x,t}}{\sqrt{2(1-y_{a_z,t})}} \end{bmatrix}^\top. \tag{3-14}$$

Algorithm 2 Tilt estimation from accelerometer readings.

Input: Normalized accelerometer data $\{y_{a,t}\}_{t=1}^{N}$.
Output: Orientation estimate $\hat{q}_t^{GS}$, where only the inclination is observable.

1: **for** $t = 1, \ldots, N$ **do**
2: **if** $y_{a_z,t} \geq 0$ **then**

$$\hat{q}_t^{SG} = \begin{bmatrix} \sqrt{\frac{y_{a_z,t}+1}{2}} & -\frac{y_{a_y,t}}{\sqrt{2(y_{a_z,t}+1)}} & \frac{y_{a_x,t}}{\sqrt{2(y_{a_z,t}+1)}} & 0 \end{bmatrix}^\top \tag{3-17}$$

3: **else**

$$\hat{q}_t^{SG} = \begin{bmatrix} -\frac{y_{a_y,t}}{\sqrt{2(1-y_{a_z,t})}} & \sqrt{\frac{1-y_{a_z,t}}{2}} & 0 & \frac{y_{a_x,t}}{\sqrt{2(1-y_{a_z,t})}} \end{bmatrix}^\top \tag{3-18}$$

4: **end if**
5: Set

$$\hat{q}_t^{GS} = \left(\hat{q}^{SG}\right)^c \tag{3-19}$$

6: **end for**

Setting q_3 equal to zero would not be of any help since this would yield another quaternion that also has a singularity at $y_{a_z,t} = -1$ as

$$\hat{q}_t^{SG} = \begin{bmatrix} \sqrt{\frac{y_{a_z,t}+1}{2}} & -\frac{y_{a_y,t}}{\sqrt{2(y_{a_z,t}+1)}} & \frac{y_{a_x,t}}{\sqrt{2(y_{a_z,t}+1)}} & 0 \end{bmatrix}^\top. \tag{3-15}$$

If we want to obtain the quaternion q^{SG} we need to create an algorithm that is able to continuously switch between two expressions for q^{SG}, so that it will not be troubled by the singularities at $y_{a_z,t} = -1$ or $y_{a_z,t} = 1$. This can be done by switching between two equations where one has the singularity at $y_{a_z,t} = -1$ and the other at $y_{a_z,t} = 1$. The final step in the tilt estimation problem is to obtain the inverse of the calculated quaternion, i.e., $\hat{q}_t^{GS}$. This is done by

$$\hat{q}_t^{GS} = \left(\hat{q}^{SG}\right)^c \tag{3-16}$$

The complete algorithm for tilt estimation with accelerometer readings only can be seen in algorithm 2. It should be noted that this algorithm uses Eq. (3-14) and Eq. (3-15). Other possibilities between equations are also possible as long as both equations do not have the same singularity.

3-2 Numerical Simulations of the Accelerometer-Only Tilt Estimation

In this section we show how Algorithm 2 performs during numerical simulations. This is done because we can create a simulated environment to research and examine the algorithms' specific properties, e.g., robustness against noise, external accelerations, and outliers. For our

simulated data-set we draw inspiration from [24]. The simulated data-set consists of a simulated Inertial Measurement Unit (IMU) that rotates around each axis for 100 samples with a sampling frequency of 10 Hz, and an initial orientation as a unit-quaternion of $\begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^\top$. The total amount of samples is chosen to be 8000. During the first 200 samples, i.e., 20 seconds, the IMU does not rotate; after this, the IMU will rotate around each axis a full 360° consecutively. We have also added noise on the gyroscope as well as on the accelerometer, where $e_{\omega,t} \sim \mathcal{N}(0, \Sigma_\omega)$ and $e_{a,t} \sim \mathcal{N}(0, \Sigma_a)$ with $\sigma_\omega = \frac{5\pi}{180}$ rad/s and $\sigma_a = -\frac{g}{100}$. During the first simulation, we will test our algorithm with this data-set. In the second simulation, we add a simulated external acceleration; were we make a crude imitation of a human moving in a straight line, with the IMU placed upon the wrist. We have done this by assuming that for one swing of the arm, it takes roughly one full second to go back and forth. Because of this swinging pattern that the arm has, we choose to simulate the external acceleration as a sinusoidal wave i.e. $a_t^G = \begin{bmatrix}\sin(2\pi \cdot T \cdot t) & \frac{1}{3}\sin(2\pi \cdot T \cdot t) & \frac{2}{3}\sin(2\pi \cdot T \cdot t)\end{bmatrix}^\top$. The external acceleration force is unevenly distributed between the x-axis, y-axis, and z-axis, this is because when the arm moves when walking in a straight line most external accelerations are in the x and z directions. For both simulated data-sets, we perform 100 Monte Carlo Simulations; and calculate the Root Means Square Error (RMSE) and the standard deviation of the error for the roll and pitch estimates. Subsequently to visualize the workings of the algorithm during the two simulations we created boxplots, which allows us to observe the median, the edges, and the whiskers of the errors that have been generated during the 100 Monte Carlo simulations. The median shows us the mean of the error, while the edges of the box are the 25^{th} and 75^{th} percentiles. The whiskers extend to the most extreme data-point that are considered to be not outliers, and the outliers are plotted individually as red crosses. The results for each corresponding simulation are to be found in Table 3-1 and the boxplots of the simulations are shown in Figure 3-1.

Simulation		Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	$\sigma_{\hat{\phi}}$	$\sigma_{\hat{\theta}}$
1	Known noise, variance and initial orientation	0.5731°	0.5729°	0.0047°	0.0042°
2	Added external sinusoidal accelerations	1.4958°	4.1739°	0.0065°	0.0066°

Table 3-1: Simulation results for tilt estimation algorithm 2, where we observe a decrease in accuracy and performance due to external accelerations, that violate the assumed accelerometer measurement model.

For our first simulation where the simulated IMU is only rotated, we observe that the tilt estimation algorithm in Algorithm 3-2 perform well. Therefore, at first glance, everything seems to work properly, but these tilt estimations did not include external accelerations. During the second simulation, we observe a clear degradation in the performance of the tilt estimation algorithm. This decrease in performance can be explained through the violation of the measurement model, which was given in Eq. (3-2). In this measurement model, we assume that no external acceleration force acts upon the IMU, clearly as explained above, we install an additional simulated external acceleration on the IMU. Thus, the assumption of the measurement model is violated, and the tilt estimation algorithm cannot compute the correct

unit-quaternion. What is also observed is that we see a difference in the increase of the total RMSE for the roll and pitch. This is due to the unequally distributed external accelerations. When an external acceleration is performed in the x direction, the resulting RMSE of the pitch will increase the most due to this acceleration. For external accelerations performed in the y direction, we observed that the RMSE of the roll would increase the most. Increased accelerations in the z direction have little effect on the accuracy of the tilt estimation. This makes sense since these accelerations are in the same direction as the gravity vector. Below in Table 3-2, an overview is given where the change in RMSE is shown when the external acceleration is distributed over the three axes. Here the sinusoidal external acceleration a_t^G as we saw previously changed, i.e. $\begin{bmatrix}\sin(2\pi \cdot T \cdot t) & 0 & 0\end{bmatrix}^\top$ is depicted as $\begin{bmatrix}1 & 0 & 0\end{bmatrix}^\top$ and $\begin{bmatrix}\sin(2\pi \cdot T \cdot t) & \frac{1}{3}\sin(2\pi \cdot T \cdot t) & \frac{2}{3}\sin(2\pi \cdot T \cdot t)\end{bmatrix}^\top$ is depicted as $\begin{bmatrix}1 & \frac{1}{3} & \frac{2}{3}\end{bmatrix}^\top$.

	$[1 \quad 0 \quad 0]^\top$	$[0 \quad 1 \quad 0]^\top$	$[0 \quad 0 \quad 1]^\top$	$[1 \quad \frac{1}{3} \quad \frac{2}{3}]$
Roll ($\hat{\phi}$)	0.5733°	4.1583°	0.5778°	1.4958°
Pitch ($\hat{\theta}$)	4.1592°	0.5716°	0.5776°	4.1739°

Table 3-2: Effects on the RMSE of the tilt estimation for different external accelerations.

In Figure 3-1, the boxplots are shown for the roll and pitch of the first and second simulation. Here, ϕ_i and θ_i with $i = 1, 2$, denote the first and second simulation for the roll and pitch respectively.

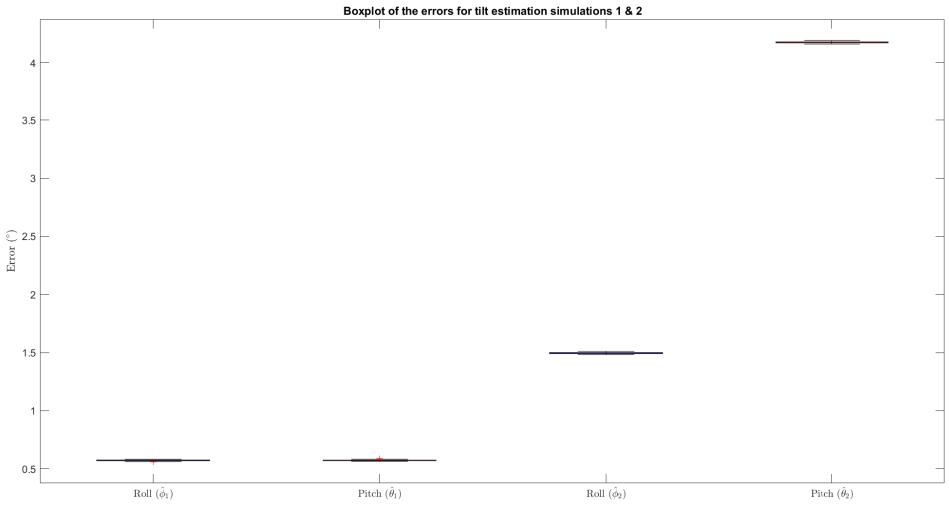


Figure 3-1: Boxplot of the first and second simulation of the tilt estimation algorithm, where in the first simulation no external acceleration is added.

It shows corresponding results with the ones that have been presented in table 3-1. Here it is visualized that due to the additional external acceleration the error increases rapidly. Due to the low values for the low values for the standard deviations in both simulations, the error values are highly condensed around the mean value, and therefore a low spread is observed.

3-3 Experimental Results of the Accelerometer-Only Tilt Estimation

For the experimental results, we will use real-life accelerometer data from the Xsens MTw Awinda; the MTw Awinda is the second generation wireless motion tracker that can track inertial and magnetic data [45, 33]. This allows us to track real-time 3D kinematic applications accurately. The Xsens' sensors can be used as a crude indicator for the validation of an estimation algorithm; this is due to Xsens' software package that can convert inertial data together with magnetometer readings to an orientation estimation [33]. The orientation estimation obtained through the MT Manager software package may serve as a reasonable ground-truth. Please note that using the orientation estimation obtained from the Xsens software as a ground-truth is not equal to using a ground-truth that can be obtained from an optical reference system. This is due to the Xsens orientation estimation error. We present our tilt estimation algorithm's findings when use is made of an Xsens MTw Awinda as follows. We performed two experiments, during the first experiment, we placed the MTw Awinda IMU upon a flat surface and slowly rotated the MTw Awinda by hand, with a sampling frequency set to 100Hz. During the second experiment, the MTw Awinda IMU was attached to the left wrist of a human, as seen in Figure 3-2, and the sampling frequency was again set equal to 100Hz.

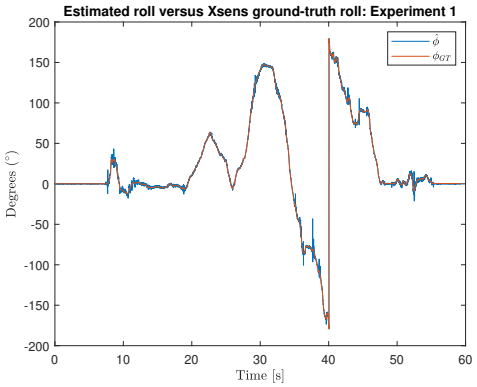
(a) Side view of the setup for tilt estimation when, the participant is walking in a straight line.

(b) Top view of the setup for tilt estimation when, the participant is walking in a straight line.

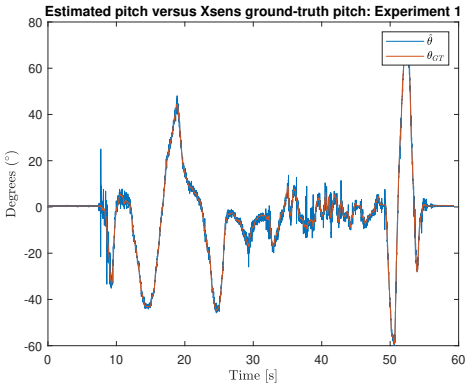
Figure 3-2: Setup for experiment two, where the IMU is attached to the left wrist.

After the sensor started recording, the subject was asked to perform a series of steps in a straight line. After the subject performed the requested movement, the sensor was removed,

and the MT Manager software package extracted the inertial data. For both experiments, we performed our tilt estimation algorithm with the accelerometer measurements. In Figure 3-3, we show our tilt estimation algorithm versus the Xsens orientation estimation when the MTw Awinda is slowly rotated. Here, ϕ_{GT} and θ_{GT} stand for the Xsens ground-truth of the roll and pitch, respectively.



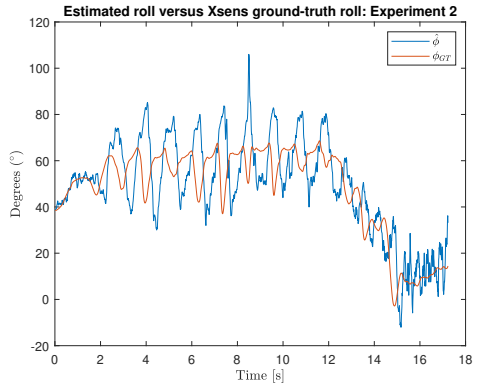
(a) Roll angle of the Xsens ground-truth versus the roll angle that is obtained via algorithm 2 for the first experiment.



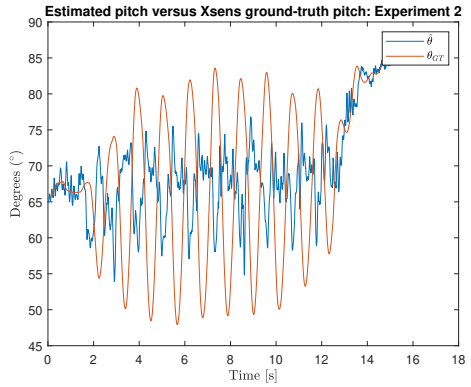
(b) Pitch angle of the Xsens ground-truth versus the pitch angle that is obtained via algorithm 2 versus the first experiment.

Figure 3-3: Results for experiment 1, where the IMU is slowly rotated.

For this experiment, an RMSE of 1.5478° and 1.9651° were obtained for the roll and pitch, respectively. This indicates that in a real setup, the algorithm performs well when slow rotations-only are performed. This was expected since this was also the case for the simulated results when no external accelerations and, thus, no movement, were added. The results of the second experiment, when the subject was asked to walk in a straight line, are found in Figure 3-4.



(a) Roll angle of the Xsens ground-truth versus the roll angle that is obtained via algorithm 2.



(b) Pitch angle of the Xsens ground-truth versus the pitch angle that is obtained via algorithm 2.

Figure 3-4: Tilt estimation with Xsens orientation estimates as ground-truth.

As can be seen in Figure 3-4, both estimation of the roll and pitch are not able to converge towards the Xsens ground-truth, an RMSE of 11.9351° and 2.0940° for the roll and pitch

were obtained respectively. The problem causing this was already discussed in the section above, where we explained how violations of the measurement model could create inaccurate inclination estimates. This is also the case for this trial. Due to the swinging of the arm, the IMU placed upon the wrist experiences external accelerations, which cause the inaccuracy in the tilt estimation. Finally, we can conclude that tilt estimation, when only an accelerometer is used, is not viable as an extension for prostheses evaluation and research.

3-4 Activity Scoring System V.1.

The use of activity counts (AC) from Actigraph as a research tool brings forth certain restrictions and limitations [16]. There has not yet been a globally accepted independent method to conduct prostheses research where activities are graded and used for further research. The use of AC is a popular method but has a dependency on the company Actigraph. The exact creation of these counts is confidential and undisclosed by the company. What is known about the definition of the activity count, is that an activity count is being defined as an acceleration signal exceeding a certain threshold where anything below this threshold is disregarded [2]. As mentioned in the discussion of [16], it would be beneficial to create a percentage contribution of each arm from the raw acceleration data; this would allow researchers compatibility with other accelerometers from different companies. In this section, we will try to create an extension as proposed by the authors of [16]. Their statement regarding the benefits of creating a method that only uses the raw accelerometer data is correct since this would eliminate the dependency on Acitgraph and the AC. However, it is not possible to determine a percentage contribution of each arm from raw accelerometer measurements. This is because when the sensor is rotated, i.e., an activity is performed, the rotating gravity vector is divided between the three independent axes, plus an additional external acceleration term is measured. A possible solution to overcome this problem could be to use the length of the accelerometer measurement vector and the gravity vector's length.

The idea is to introduce a new way to create the vector magnitude (VM) and create a novel scoring system based on the participant's acceleration when a high-intensity movement pattern with the arms is performed e.g., cooking. This has the benefit that the scoring system is custom-made for each participant since a high-intensity movement pattern differs for each person. Another benefit of the scoring system could be that we can understand all activities' intensity in a much better way. For example, single-use of the dominant arm is observed whenever the participant performs an activity with only the dominant arm. This could either be a low-intensity activity or a high-intensity activity; both will always be seen as the same type of activity, namely unilateral for the dominant hand. The scoring system introduces an additional measure to comment on the intensity of the activities performed. The scoring system consists of four parts; the determination of the high-intensity activity, the determination if the VM equals zero, the calculation of the VM and activity score per measurement, and the conversion into epochs. The high-intensity activity, $\mathrm{VM}_{a,high}$, is defined as a VM, which tells something about the amount of acceleration that the participant can generate with the dominant or anatomically intact (AI) arm when a high-intensity activity is performed. This can be easily performed at the beginning of the trial by asking the participant to perform a set of tasks that require intense use of the hands. The VM for the acceleration

per measurement can be calculated as

$$\mathrm{VM}_{a,t} = \left| \|y_{a,t}\|_2 - 9.81 \right| \tag{3-20}$$

The 2-norm of the gravity vector is subtracted from the length of the accelerometer measurement vector. With this, we create a scalar value that indicates how much external acceleration was delivered for each measurement. This method is only used for the cases when movement occurs i.e., external accelerations occur. This is because Eq. (3-20), does not take into account the sensor noise that is induced upon the accelerometer signal. To determine whether no movement occurs and thus when the VM should equal zero, we make use of the following assumption that when the sensor does not move, the following approximation exists as

$$\|y_{a,t}\|_2 - \|g\|_2 = \|y_{a,t}\|_2 - 9.81 \approx 0 \tag{3-21}$$

By assuming that Eq. (3-21) is zero-mean Gaussian distributed, we can find the variance and standard deviation. After this, we can examine if the results of Eq. (3-21) fall between a specific bound created out of a chosen amount of standard deviations. This entails that the higher we set the bounds, the higher the chance that the signal will be classified as zero. To create a scoring system that shows us the intensity of the use in an arm, we rank the VM that is obtained in Eq. (3-20). This is done by calculating the ratio between the VM of Eq. (3-20) against the VM that was obtained when a high-intensity activity was performed as follows

$$\mathrm{S}_{a,t} = 100 \cdot \frac{\mathrm{VM}_{a,t}}{\mathrm{VM}_{a,high}}. \tag{3-22}$$

With the VM and activity scores obtained from Eq. (3-20) and Eq. (3-22), it is possible to create epochs; this is done to drastically decrease the computational intensity for any other further research that may be conducted. Epochs are created by first choosing the length of the epochs, for example, if one chooses the epoch length, l_e to be two seconds and the sensor measures with a frequency of 10 Hz, one epoch consists of the scalar value, which can be obtained by taking the average of twenty consecutive VM. The general equation for this is as

$$\mathrm{VM}_{a,e} = T \cdot \frac{1}{l_e} \left(\mathrm{VM}_{a,t} + \ldots + \mathrm{VM}_{a,t+\frac{l_e}{T}} \right), \tag{3-23}$$

similarly, this can be done for the scoring system as

$$\mathrm{S}_{a,e} = T \cdot \frac{1}{l_e} \left(\mathrm{S}_{a,t} + \ldots + \mathrm{S}_{a,t+\frac{l_e}{T}} \right). \tag{3-24}$$

Like [16], we can create a percentage-wise contribution with the epochs from the VM as

$$\mathrm{Contribution}_{\mathrm{Dom}} = \left[\text{ round } \left(\frac{\mathrm{VM}_{D,a,e}}{\mathrm{VM}_{D,a,e} + \mathrm{VM}_{ND,a,e}} \cdot 100 \right) \right], \tag{3-25}$$

and for the non-dominant

$$\mathrm{Contribution}_{\mathrm{Non-Dom}} = \left[\text{ round } \left(\frac{\mathrm{VM}_{ND,a,e}}{\mathrm{VM}_{D,a,e} + \mathrm{VM}_{ND,a,e}} \cdot 100 \right) \right], \tag{3-26}$$

here $VM_{D,a,e}$ is the value from epoch e, from the VM of the dominant arm, and $VM_{ND,a,e}$ for the non-dominant arm. The percentage-wise contribution has already been proven to be an effective tool for prosthesis research, as was seen in [16]. Therefore, we will also try to create it with our proposed method.

3-5 Experimental Results of the Activity Scoring System V.1.

This section will show how our proposed method works, when it is tested on a real data-set. The data-set was obtained by wearing the AX6 monitor from Axivity for one day, with the participant's right hand being dominant. The AX6 is capable of measuring the acceleration and angular velocity for up to seven days. During the day that the measurements were conducted, two Axivity AX6 monitors were worn, one on each wrist. The sensors are mounted as prescribed by Axivity, the USB port of the sensor placed on the right wrist is pointing towards the ground, while the USB port of the sensor placed on the left wrist is pointing upwards to the sky. The sensors' placement is assisted with a silicone wrist band, which provides a convenient way of mounting the sensors on the wrist, as can be seen in Figure 3-5.

(a) Top view of the two AX6 monitors.

(b) Left side view of the AX6 monitor.

Figure 3-5: Setup for the 24 hour experiment, where two AX6 Axivity monitors are attached to the wrists of the participant.

The sensors were both set to measure at a sampling rate of 100 Hz for 24 hours. After the data was collected, we corrected the data-sets of the right and left arm such that the time instances correspond with each other. Subsequently, we examined every time instance with the help of the proposed method that uses Eq. (3-21) to determine if the sensor and thus, the arms were not moving. To do so, we manually found a string of data when the arms were not moving and extracted this data to determine the standard deviation as proposed previously. After this, we found a value of $\sigma_a = 0.0056$ m/s^2, which we used for both sensors. To determine the high-intensity activity VM, we created a second data-set, in which the participant performs

a constant movement pattern of the arms, where the participant rolls a ball on a table and tosses that ball from one hand to the other as can be seen in Figure 3-6.

(a) The rolling of the ball, which is a high intensity movement of the arms

(b) Tossing the ball from the left hand to the right hand.

Figure 3-6: The creation of the data-set, from which the high intensity VM is determined.

After this a $\text{VM}_{a,high} = 0.5690$ m/s^2 is found. Our first trial uses a bound of $\pm 3\sigma_a$, which, more formally speaking, uses the three-sigma rule to identify the percentage of values that lie inside a band around the mean in a normal distribution with a width of six standard deviations; this means that 99.73% of the values lie within three standard deviations of the mean [48]. After the determination of the VM that are zero, the non-zero VM are calculated with Eq. (3-20). In this work, we will only consider one-second epochs; this is a relatively short epoch time, but necessary to ensure the scoring system's accuracy. If the epoch time is taken too large, we enhance computational efficiency but are prone to lose valuable information about the intensity of the performed movement. This is because, during a longer epoch, there may be very short outbursts of intense use of the arms, which will be lost due to the averaging of the scores required to create an epoch.

Similar to [16], whenever the VM of both arms was equal to zero, it was removed from the data-set. After this, percentage bands from 0-100% with an increment of 1% were created, and the time in seconds was added up. We display the histograms in the same fashion as the authors from [16], as seen in Figure 3-7.

Comparing Figure 3-7, with Figure 2-2, which was the contribution histogram from [16], we observe similar results regarding the shape of the histogram. This could be an indication that the creation of the VM with the six standard deviation bounds work. Like [16], we can also create an Archimedean spiral plot but incorporate the scoring system as a second larger spiral underneath the first spiral, which indicates how the arms are used. This allows us to observe patterns in the use of the arms, with an additional observation regarding the intensity of that use.

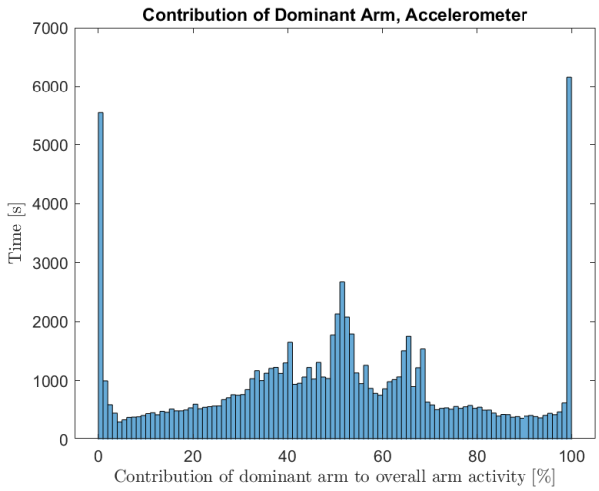


Figure 3-7: Histogram plot of the contribution of the right arm, when two AX6 Axivity monitors are worn for 24 hours. The histogram has a similar shape as the ones that are published in [16].

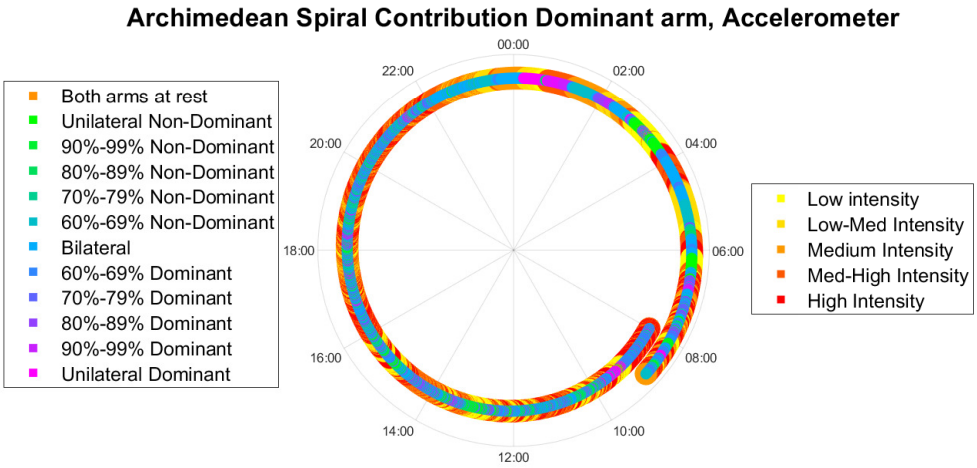


Figure 3-8: The two Archimedean spirals that are created during the first experiment, where we use a three standard deviations upper and lower bound. We observe that from this we are not able to obtain a correct spiral for the use of hands.

From Figure 3-8, we observe that the spiral plots are not correct. During the night, when the participant sleeps, we should predominantly observe "Both arms at rest". However, we observe either bilateral, unilateral non-dominant, and unilateral dominant movements. The reason for this is the low values for the bound, which determines whether the VM is classified as zero. What occurs is that during the creation of an epoch, when the sensor is not moving, one or more VM is wrongly classified as non-zero. This results in the whole epoch being classified as a non-zero epoch. This also explains why we only observe bilateral and unilateral movement patterns during the night.

To overcome this problem, we can increase the bound's value, such that the VM are classified as zero more often. We experimentally found a bound of $\pm 12\sigma_a$, which gave the following

results for the histogram as can be seen in Figure 3-9

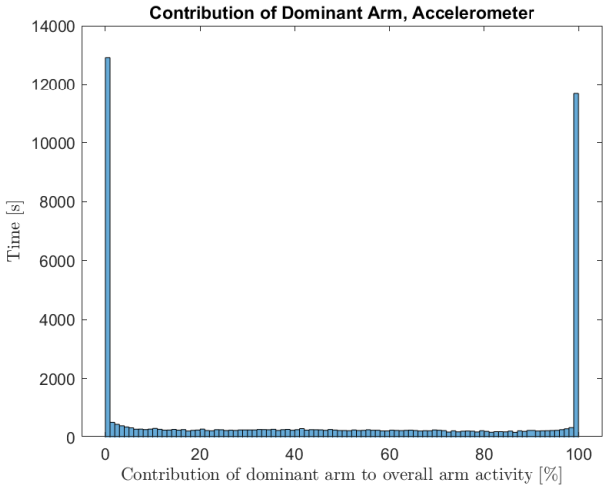


Figure 3-9: The results of the histogram for the second experiment where we increased the the value for the bounds. Almost all bilateral activity is removed and an increase in unilateral movement observed due to the wrong classification.

Due to the increase of the standard deviation bound, the histogram is not reliable anymore. We observe that most low intense bilateral activities are now being classified as a VM of zero. Therefore, we see an increase in both unilateral movements, which in Figure 3-7 were roughly 5000-6000 seconds each, and have now increased to 12000-13000 seconds. The corresponding spiral plot is shown below in Figure 3-10

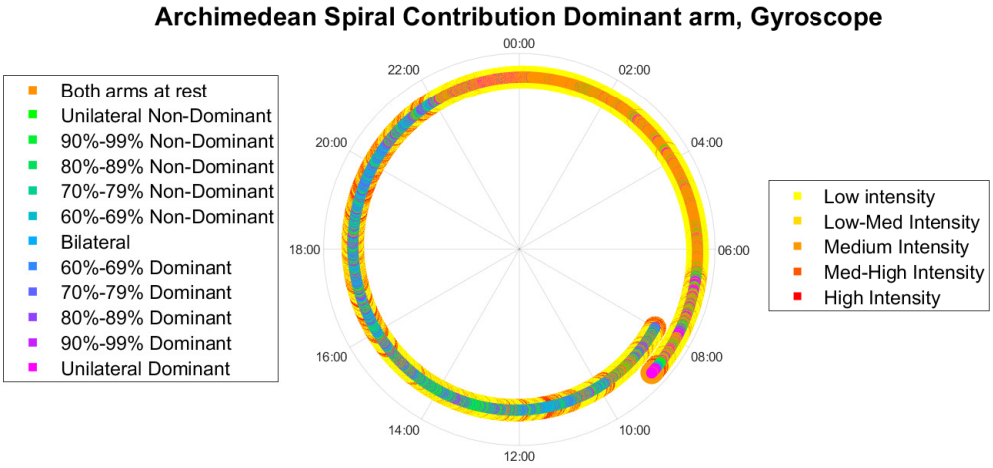


Figure 3-10: The two Archimedean spirals for experiment two where the bounds are increased. We observe that both spirals are correct. This is due to the increased value for the bounds, which causes a more accurate classification of the moments when both arms are at rest.

Unlike the spiral plot from Figure 3-8, we are able to obtain a correct spiral plot, from which we can observe when the participant is awake, when the participant is sleeping and how the

arms are used during the day. We can also observe that there are frequent outbursts of high-intensity use of the dominant arm during the day. From this we can conclude that the choice regarding the bounds, creates a trade off between the correct classification of no movement, or the correct classification of low intense movement, which cause low acceleration signals to occur.

Chapter 4

Prosthesis analysis with a Single Inertial Measurement Unit Per Arm

In this chapter, we will discuss the benefits of the second option regarding the sensors to improve upper artificial limb analysis. Here we introduce the gyroscope as an additional sensor that can be used. In Chapter 3, it was clear that we are not able to obtain a stable tilt estimate due to violations that occur while moving. In this chapter, using a gyroscope enables us to use more advanced Sensor Fusion (SF) techniques to acquire correct information about the sensor's inclination. Before estimating the sensor's inclination, an estimation algorithm of the initial inclination is required. After this, we will discuss how the complementary filter can create a drift-free tilt estimate, and finally, an extension of the Activity Scoring System will be introduced. Similar to Chapter 3, we will use simulated data as well as experimental data-sets, and for the extension on the Activity Scoring System, we use the data-set obtained from wearing the Axivity AX6 Inertial Measurement Unit (IMU).

4-1 Initial Tilt Estimation via the Triad Method

In this section, we will show how an estimate of the initial inclination can be created by altering the TRIAD algorithm shown in Section 2-2-3. As mentioned before, the TRIAD method utilizes the first magnetometer and accelerator measurements. Unless the measurements are taken at the magnetic north or south poles, the first measurements will contain information about the orientation [23]. Normally, four different vectors are obtained, the accelerometer measurement $y_{a,t}$, the magnetometer measurement $y_{m,t}$, the local gravity vector g^G and the local magnetic field vector m^G. In this version of the TRIAD method, no magnetometer measurements will be available. Therefore, we require a solution to the absence of magnetometer measurements. The solution to this problem is rather straight forward. We can remove every part that has to do with the magnetometer, i.e., remove it from the objective function. For the accelerometer we then obtain

$$y_{a,t} = \begin{bmatrix} y_{a_x,t} & y_{a_y,t} & y_{a_z,t} \end{bmatrix}^\top, \quad \|y_{a,t}\|_2 = 1 \quad \forall\, t, \tag{4-1}$$

$$\hat{g}^G = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^\top, \quad \hat{g}^S = \frac{y_{a,1}}{\|y_{a,1}\|_2}. \tag{4-2}$$

The optimization problem for the initial inclination minimizes the distance between the gravity vector measured at the first time step and the normalized gravity vector in the global frame as

$$\begin{aligned} \underset{q^{GS}}{\arg\min} \quad & \left\| \bar{\hat{g}}^G - q^{GS} \odot \bar{\hat{g}}^S \odot q^{SG} \right\|_2^2 \\ \text{subj. to} \quad & \left\| q^{GS} \right\|_2 = 1. \end{aligned} \tag{4-3}$$

In Eq. (4-3), quaternion multiplications are performed between the orientation estimate and the gravity vector. These multiplications can only be performed if the gravity vector in both sensors and the global frame is in $\mathbb{R}^{4\times 1}$. The authors of [31] propose a simple conversion method for vectors that inherently are not four-dimensional. The vector can be extended by adding an additional zero element, which will increase the dimension of the vector while still keeping it at unit length, e.g, $\bar{\hat{g}} = \begin{bmatrix} 0 & \hat{g}^\top \end{bmatrix}^\top$. With this conversion, the matrix, A, can be created as

$$A = -\sum_{n=1}^{N} \left(\bar{\hat{g}}_n^G\right)^L \left(\bar{\hat{g}}_n^S\right)^R = -\left(\bar{\hat{g}}^G\right)^L \left(\bar{\hat{g}}^S\right)^R. \tag{4-4}$$

Here, the superscripts L and R stand for the left and right quaternion multiplication, as defined in Eq. (2-14). With this matrix A, the optimization problem to find the initial inclination can be rewritten as

$$\begin{aligned} \hat{q}_1^{GS} = \arg\min_{q^{GS}} \quad & \left(q^{GS}\right)^\top A q^{GS}, \\ \text{subj. to} \quad & \left\| q^{GS} \right\|_2 = 1. \end{aligned} \tag{4-5}$$

The optimization problem is solved for setting $\hat{q}^{GS} = v_1$, where v_1 is the eigenvector corresponding to largest positive eigenvector λ_1 of the system $Av = \lambda v$ [21]. This can be proven as follows when we write the squared residual as

$$\|e_n\|_2^2 = \left\|\bar{\hat{g}}_n^G\right\|_2^2 - 2\bar{\hat{g}}_n^G \cdot \left(q^{GS} \odot \bar{\hat{g}}_n^S \odot q^{SG}\right) + \left\|\bar{\hat{g}}_n^S\right\|_2^2 \tag{4-6}$$

The minimization of the objective function can only affect the middle term; therefore we can simplify it as

$$\begin{aligned}\bar{\hat{g}}_n^G \cdot \left(q^{GS} \odot \bar{\hat{g}}_n^S \odot q^{SG}\right) &= -\left(\bar{\hat{g}}_n^G \odot \left(q^{GS} \odot \bar{\hat{g}}_n^S \odot q^{SG}\right)\right)_0 \\ &= -\left(\bar{\hat{g}}_n^G \odot q^{GS}\right)^\top \left(\bar{\hat{g}}_n^S \odot q^{SG}\right)^c \\ &= -\left(q^{GS}\right)^\top \left(\bar{\hat{g}}_n^G\right)^L \left(\bar{\hat{g}}_n^S\right)^R q^{GS}\end{aligned} \tag{4-7}$$

By using the relation that $(a \odot b)_0 = a^\top b^c$ for the scalar part of quaternion multiplication, we can reformulate the objective function as

$$\underset{\|q^{GS}\|_2=1}{\arg\min} \sum_{n=1}^{N} \|e_n\|_2^2 = \underset{\|q^{GS}\|_2=1}{\arg\max} \left(q^{GS}\right)^\top A q^{GS}. \tag{4-8}$$

Here the matrix A is defined as Eq. (4-4). By using Eq. (C-26), Eq. (C-30) and Eq. (C-31) we create

$$\left(\bar{\hat{g}}_n^G\right)^L \left(\bar{\hat{g}}_n^S\right)^R = \left[-\left(\bar{\hat{g}}_n^G\right)^L\right]^\top \left[-\left(\bar{\hat{g}}_n^S\right)^R\right]^\top = \left[\left(\bar{\hat{g}}_n^S\right)^R \left(\bar{\hat{g}}_n^G\right)^L\right]^\top = \left[\left(\bar{\hat{g}}_n^G\right)^L \left(\bar{\hat{g}}_n^S\right)^R\right]^\top. \tag{4-9}$$

From this we can conclude that A is a real symmetric matrix, i.e $A = A^\top$. Let $q^{GS} = V\alpha$ with $\|\alpha\|_2 = 1$, where V is an orthonormal basis obtained from the symmetric eigenvalue decomposition of $A = V\Lambda V^\top$. Then,

$$\left(q^{GS}\right)^\top A q^{GS} = \alpha^\top V^\top V \Lambda V^\top V \alpha = \sum_{i=1}^{4} \alpha_i^2 \lambda_i \leq \lambda_1 \tag{4-10}$$

where, λ_1 is the largest positive eigenvalue of A. Equality is obtained for $\alpha = [1 \quad 0 \quad 0 \quad 0]^\top$, that is, $\hat{q}^{GS} = v_1$.

4-2 Tilt Estimation with a Complementary Filter

In Section 3-1, we observed how the use of only an accelerometer is not suitable for tilt estimation in prosthesis research and evaluation, this was due to the frequent violations of the measurement model that occur during movement of the arm(s), which induced significant errors regarding the accuracy of the estimate. In this section, we will present a method that can compensate for the violation of the measurement model of Eq. (2-8) and is computationally very efficient. This SF method will fuse information of the gyroscope and the accelerometer; a complementary filter will assist the fusion of information. Most complementary filters use a magnetometer, like the one shown in Section 2-2-4 from [24]. For upper-limb prostheses analysis, a magnetometer will not be available, due to the likely encounters with a ferromagnetic material that will cause magnetic disturbances [23]. Hence a complementary filter is required that does not require a magnetometer. Below, we show the changes made with respect to the complementary filter presented in Section 2-2-4 from [24].

Since no magnetometer is used in this problem, we can reformulate the estimation problem. The orientation estimation from accelerometer measurements can be formulated as an optimization problem where we use the measurement model of Eq. (2-8) as

$$\min_{\eta_t} V(\eta_t) = \min_{\eta_t} \frac{1}{2} \left\| y_{a,t} + \left(\exp_{\mathrm{R}}(\eta_t)\right)^{\top} R\left(\tilde{q}_t^{SG}\right) g^G \right\|_2^2. \tag{4-11}$$

The change in the objective function in Eq. (4-11), will result in the following derivative as

$$\nabla V(\eta_t) = -\left[R\left(\hat{q}_{t-1}^{SG}\right) g^G \times\right] \left(y_{a,t} + R\left(\hat{q}_{t-1}^{SG}\right) g^G\right). \tag{4-12}$$

The derivative can be computed as

$$\nabla V(\eta_t) = \frac{\partial V(\eta_t)}{\partial \eta_t} = \left(\frac{\partial e_t}{\eta_t}\right)^{\top} e_t. \tag{4-13}$$

Here $e_t = y_{a,t} + \left(\exp_R(\eta_t)\right)^{\top} R\left(\tilde{q}_t^{SG}\right) g^G$. The first step is the creation of $\frac{\partial e_t}{\partial \eta_t}$, which will later be transposed. The first part is to asses and work out the following

$$\begin{aligned}
e_t &= y_{a,t} + \left(\exp_R(\eta_t)\right)^{\top} R\left(\tilde{q}_t^{SG}\right) g^G \\
&= y_{a,t} + \left(\mathcal{I}_3 + [\eta_t \times]\right)^{\top} R\left(\tilde{q}_t^{SG}\right) g^G \\
&= y_{a,t} + \begin{bmatrix} 1 & -\eta_{3,t} & \eta_{2,t} \\ \eta_{3,t} & 1 & -\eta_{1,t} \\ -\eta_{2,t} & \eta_{1,t} & 1 \end{bmatrix}^{\top} \begin{bmatrix} R_{11}\left(\tilde{q}_t^{SG}\right) & R_{12}\left(\tilde{q}_t^{SG}\right) & R_{13}\left(\tilde{q}_t^{SG}\right) \\ R_{21}\left(\tilde{q}_t^{SG}\right) & R_{22}\left(\tilde{q}_t^{SG}\right) & R_{23}\left(\tilde{q}_t^{SG}\right) \\ R_{31}\left(\tilde{q}_t^{SG}\right) & R_{32}\left(\tilde{q}_t^{SG}\right) & R_{33}\left(\tilde{q}_t^{SG}\right) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\
&= y_{a,t} + \begin{bmatrix} 1 & \eta_{3,t} & -\eta_{2,t} \\ -\eta_{3,t} & 1 & \eta_{1,t} \\ \eta_{2,t} & -\eta_{1,t} & 1 \end{bmatrix} \begin{bmatrix} R_{13}\left(\tilde{q}_t^{SG}\right) \\ R_{23}\left(\tilde{q}_t^{SG}\right) \\ R_{33}\left(\tilde{q}_t^{SG}\right) \end{bmatrix} \\
&= y_{a,t} + \begin{bmatrix} R_{13}\left(\tilde{q}_t^{SG}\right) + \eta_{3,t} R_{23}\left(\tilde{q}_t^{SG}\right) - \eta_{2,t} R_{13}\left(\tilde{q}_t^{SG}\right) \\ -\eta_{3,t} R_{13}\left(\tilde{q}_t^{SG}\right) + R_{23}\left(\tilde{q}_t^{SG}\right) + \eta_{1,t} R_{33}\left(\tilde{q}_t^{SG}\right) \\ \eta_{2,t} R_{13}\left(\tilde{q}_t^{SG}\right) - \eta_{1,t} R_{23}\left(\tilde{q}_t^{SG}\right) + R_{33}\left(\tilde{q}_t^{SG}\right) \end{bmatrix}
\end{aligned}$$

It is now possible to create the derivative with respect to η_1 as

$$\frac{\partial e_t}{\partial \eta_t} = \begin{bmatrix} 0 & -R_{33}\left(\tilde{q}_t^{SG}\right) & R_{23}\left(\tilde{q}_t^{SG}\right) \\ R_{33}\left(\tilde{q}_t^{SG}\right) & 0 & -R_{13}\left(\tilde{q}_t^{SG}\right) \\ -R_{23}\left(\tilde{q}_t^{SG}\right) & R_{13}\left(\tilde{q}_t^{SG}\right) & 0 \end{bmatrix} = \left[R\left(\tilde{q}_t^{SG}\right) g^G \times\right]$$

if we substitute this in Eq. (4-13) then $\nabla V(\eta_t)$ becomes

Algorithm 3 Tilt estimation using a complementary filter with orientation deviation states.

Input: Normalized accelerometer & gyroscope data $\{y_{a,t}, y_{\omega,t}\}_{t=1}^{N}$, Initial Orientation $\hat{q}_1^{GS}$. tuning parameter β and the sampling time T.

Output: Orientation estimate $\hat{q}_t^{GS}$, from which we can only observe the inclination.

1: **for** $t = 2, \ldots, N$ **do**

$$\nabla V(\eta_t) = -\left[R\left(\hat{q}_{t-1}^{SG}\right) g^G \times\right]\left(y_{a,t} + R\left(\hat{q}_{t-1}^{SG}\right) g^G\right)$$

with $y_{a,t}$ and $\hat{q}_{t-1}^{GS}$

2: Based on $\nabla V(\eta_t)$, compute

$$\hat{\omega}_t = y_{\omega,t} - \beta \frac{\nabla V(\eta_t)}{\|\nabla V(\eta_t)\|}$$

3: Obtain the updated orientation

$$\hat{q}_t^{GS} \approx \hat{q}_{t-1}^{GS} + \frac{T}{2} S\left(\hat{q}_{t-1}^{GS}\right) \hat{\omega}_t,$$

with $S\left(\hat{q}_{t-1}^{GS}\right)$ as defined in Eq. (2-27).

4: **end for**

$$\begin{aligned}
\nabla V(\eta_t) &= \left[R\left(\hat{q}_{t-1}^{SG}\right) g^G \times\right]^\top \left(y_{a,t} + \left(\exp_R(0)\right)^\top R\left(\tilde{q}_t^{SG}\right) g^G\right) \\
&= -\left[R\left(\hat{q}_{t-1}^{SG}\right) g^G \times\right]\left(y_{a,t} + \left(\mathcal{I}_3 + [0\times]\right)^\top R\left(\tilde{q}_t^{SG}\right) g^G\right) \\
&= -\left[R\left(\hat{q}_{t-1}^{SG}\right) g^G \times\right]\left(y_{a,t} + R\left(\hat{q}_{t-1}^{SG}\right) g^G\right)
\end{aligned}$$

The estimate of the orientation deviations state can now be used to compute $q_{a,t}^{GS}$ instead of $q_{am,t}^{GS}$ of Eq. (2-33), since here the algorithm only relies on the accelerometer measurements. The final algorithm of the complementary filter using orientation deviation states for tilt estimation is given in Algorithm 3.

4-3 Numerical Simulations of the Single Inertial Measurement Unit Analysis

In this section we will perform the numerical simulations for the single sensor case, when an additional gyroscope is being used. Here we will test the initial inclination estimation algorithm that is presented in Section 4-1 and complementary filter, that is presented in Section 4-2.

4-3-1 Simulation Results of the Initial Tilt Estimation Algorithm

To analyze this TRIAD method's performance, we alter the data-set, that was used in Section 3-2, by randomly creating different initial inclinations in the form of a unit quaternion. We

then test the algorithms performance by using two simulation, where each simulation is 100 Monte Carlo simulations long. During the first simulation no added external accelerations are added, while in the second simulation we do add external accelerations, to mimic the motion of a swinging arm, as shown in Section 3-2.

After 100 Monte Carlo simulations, the Root Means Square Error (RMSE) values and standard deviations were calculated and displayed in Table 4-1, accompanied by a boxplot to visualize the results. When no external accelerations are added, the initial estimation algorithm performs well, but when the external accelerations are added during the second simulation, we observe the same type of degradation in performance due to violations of the assumption of the measurement model, as we observed earlier in Section 3-2.

Simulation		Roll ($\hat{\phi}$)	Pitch($\hat{\theta}$)	$\sigma_{\hat{\phi}}$	$\sigma_{\hat{\theta}}$
1	No added external sinusoidal accelerations	0.5538°	0.5591°	0.5528°	0.5612°
2	Added external sinusoidal accelerations	1.1642°	3.3084°	0.5320°	0.5328°

Table 4-1: Simulation results for the initial inclination estimation via the TRIAD method, where we observe a clear degradation in performance due to the accelerometer measurement model violations.

In Figure 4-1, we show the boxplots that correspond to the two Monte Carlo simulations for the initial tilt estimate. With ϕ_i and θ_i for $i = 1, 2$, which denotes the first and second simulation for the roll and pitch respectively. Here we observe a large spread, which corresponds to the high standard deviation values. This large spread, and thus high standard deviation values are likely due to the noise that is simulated upon the accelerometer measurements.

4-3-2 Simulation Results of the Tilt Estimation with a Complementary Filter

To analyze the complementary filter, we use the simulated data-set presented in Section 3-2. We will run three different simulations, where during the first simulation we have prior knowledge about the noise and the initial orientation. In the second simulation we will add an additional external acceleration on the accelerometer measurement, that mimics the movement of a swinging arm whilst walking, as we saw in Section 3-2, this allows us to create a comparison between the algorithm used in Section 3-2 and the complementary filter, when the assumed accelerometer measurement model is violated. During the third simulation, we will add another possible disturbance that may occur, in the form of accelerometer outliers. In the real world, these can be caused by the sensor or the body that the sensor is placed upon banging or ticking against a hard surface. An example of this would be that when the IMU is placed on the wrist of the anatomically intact (AI) arm or prosthesis, and the participant bumps into something with his/her arm. This will generate a spike in the accelerometer signal, which can be categorized as an accelerometer outlier. In the third simulation, we simulated these types of outliers and the external accelerations to test the robustness of algorithm

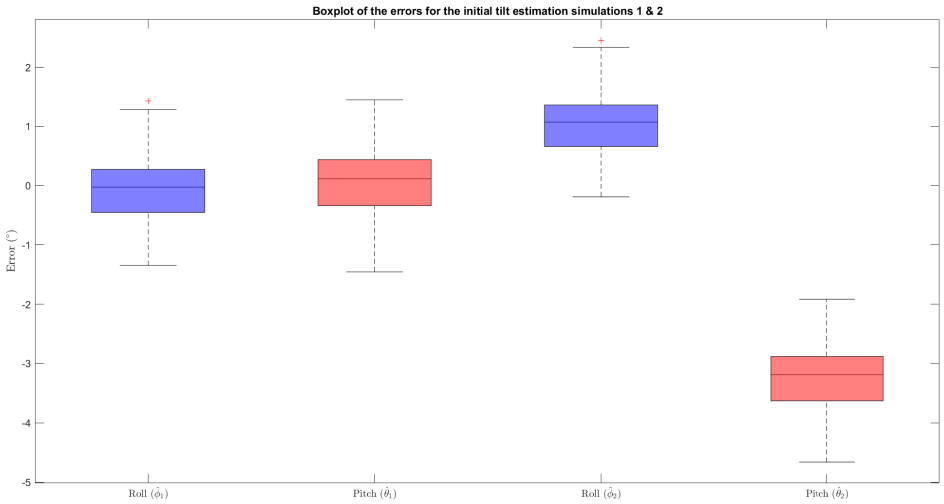


Figure 4-1: Boxplot of the first and second simulation, where 100 Monte Carlo simulations are performed for the initial tilt estimation algorithm. In the first simulation, no external acceleration are added, and during the second simulation external acceleration are added. The degradation in performance is clearly visible during the second simulation, where violations of the assumed accelerometer measurement model occur.

3. The outliers are created by replacing a percentage of the normalized accelerometer data with outliers. The outliers are chosen to be sampled from a Gaussian distribution with a covariance equal to an identity matrix. In this example, we choose to replace 5% of the normalized accelerometer data, as was similarly done in [24]. Lastly, we will also create an analysis of the convergence of the algorithm. from this, we can prove that the algorithm can converge from a wrong initial tilt towards a correct inclination estimate. In all simulations β is chosen in a similar fashion as in [24], i.e., $\beta = \sqrt{3}\sigma_\omega$. For each simulation, 100 Monte Carlo simulations are performed. The results are summarized in table 4-2, which give the RMSE and standard deviation values for the roll and pitch per simulated extension, where each column is named after the addition in the simulation upon the previous column.

After performing the simulation with known noise, variances, and initial orientation, we obtained satisfactory results, which correspond to the values that were obtained in [24], when a similar simulation was performed. After the second simulation, we observed that the error in the inclination estimation increases, this is due to the external accelerations, but we can still observe a major decrease in RMSE values when comparing the second simulation in this section with the second simulation in Section 3-2, and thus we can conclude that the use of an complementary filter is a more suitable technique for tilt estimation when conducting prostheses research. For the third simulation, we observed that the RMSE increased just the slightest when outliers were added, and thus we see that Algorithm 3 is robust against accelerometer outliers. To create a visual interpretation of the results, a boxplot combining all three simulations is given in Figure 4-2. With ϕ_i and θ_i for $i = 1, 2, 3$, which denotes the first, second and third simulation for the roll and pitch respectively.

Simulation		Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	$\sigma_{\hat{\phi}}$	$\sigma_{\hat{\theta}}$
1	Known noise, variance and initial orientation	0.7337°	0.7317°	0.0063°	0.0078°
2	Added external sinusoidal accelerations	0.9851°	1.7388°	0.0123°	0.0241°
3	5% outliers magnitude $\mathcal{N}(0, \mathcal{I})$	1.0346°	1.7806°	0.0176°	0.0287°

Table 4-2: Effects on the RMSE and standard deviation of the error for the tilt estimation with a complementary filter per simulation. In all three simulations Algorithm 3 is able to generate a correct estimate.

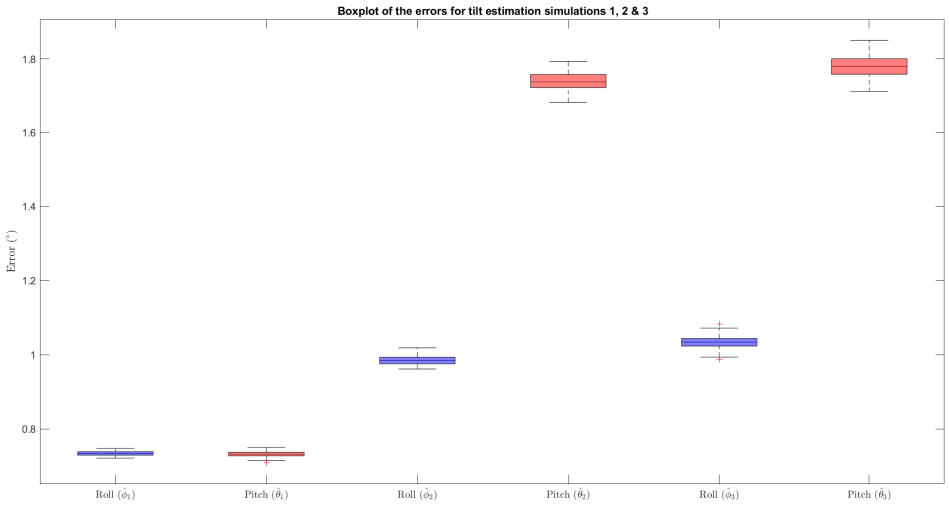


Figure 4-2: Boxplot of the first, second and third simulation of the tilt estimation algorithm, where in the first simulation no external acceleration is added, in the second external acceleration is added, and in the third additional accelerometer outliers are added. We observe an increase in RMSE, due to the external accelerations, but the tilt estimation algorithm remains accurate.

From Figure 4-2, we can see that the errors increase when the measurement model assumptions are violated. Simultaneously, the external accelerations will also increase the spread of the error. We observe the spread in the pitch values are more significant than in the roll values; this makes sense since, on this angle, the external acceleration has the most effective as we saw in Table 3-2 of Section 3-2. When we add additional accelerometer outliers, we observe little degradation in the performance of the estimation algorithm. From this, we can conclude that Algorithm 3 has excellent robustness against outliers and external accelerations, i.e., measurement model violations. The last thing that we will test is if the algorithm can generate an estimate that converges over time when a wrong initial inclination is chosen together with external accelerations and outliers. This is tested by setting the initial inclination as a wrong

arbitrary initial inclination. During this test, an initial inclination of $81.8699°$ and $-19.4712°$ is chosen for the roll and pitch angles. We perform 100 Monte Carlo simulations and calculate the mean of every iteration; simultaneously, we calculate the standard deviation per iteration. In Figure 4-3, we show the mean of the first 200 samples of the 100 Monte Carlo samples, here the shaded areas are the spread, which is given by three standard deviations. This enables us to capture 99.73% of all data that was created during this simulation.

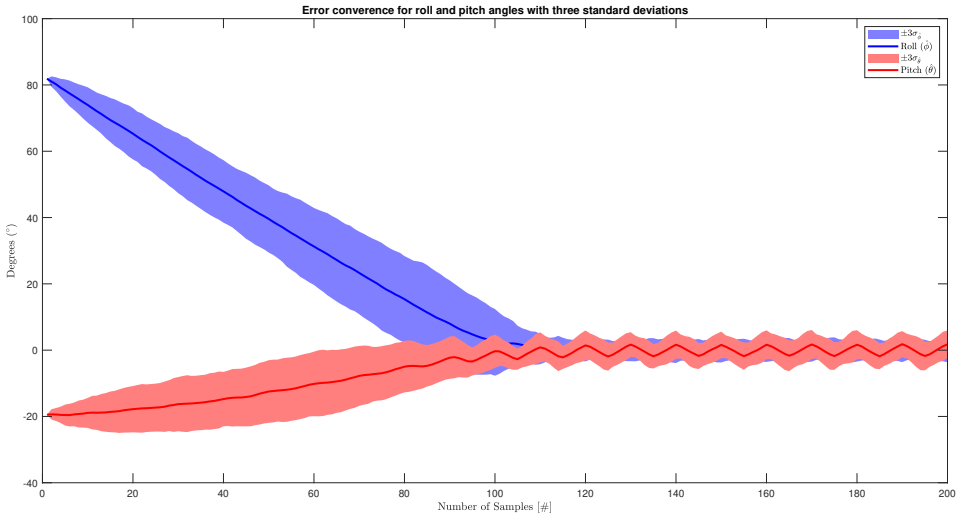


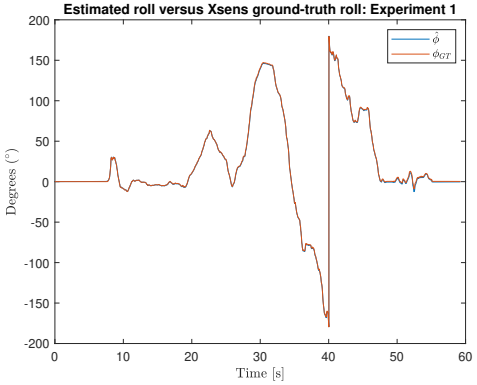
Figure 4-3: Estimation error for the first 200 samples of 100 Monte Carlo simulations with a fixed wrong initial tilt as $81.8699°$ and $-19.4712°$ for the roll and pitch respectively.

We observe an apparent convergence towards zero, but once it has reached convergence around zero, the estimation error starts to fluctuate around zero degrees error. This is due to the noise, measurement model violations, and the outliers that we have induced upon the simulated IMU. From the combined results obtained above, we can conclude that the tilt estimation algorithm presented in Algorithm 3 is robust against noise, outliers, measurement model violations, and can converge from a wrong initial inclination.

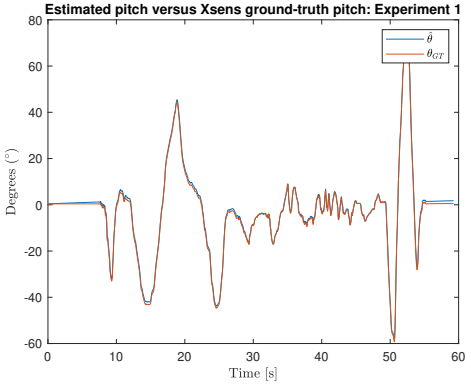
4-4 Experimental Results of the Tilt Estimation with a Complementary Filter

Similar to Section 3-2, we will use the data-sets obtained from the Xsens MTw Awinda for the experimental results. As mentioned before, during the first experiment, we placed the MTw Awinda IMU upon a flat surface and slowly rotated the MTw Awinda by hand. In the second experiment, we asked a participant to wear the MTw Awinda around the left wrist and perform a series of steps in a straight line. During both experimental setups the sampling time of the MTw Awinda sensor was set to 100 Hz. To obtain a reasonable value for β in both experiments, we calculated the standard deviation, σ_ω, by taking the first 6 seconds of the first experiment since here the sensor is lying still on a flat surface. eventually we found a value of $\beta = 2.6 \cdot 10^{-3}$. In Figure 3-3, we show the results of the tilt estimation via the

complementary filter versus the Xsens orientation estimation when the MTw Awinda is slowly rotated.



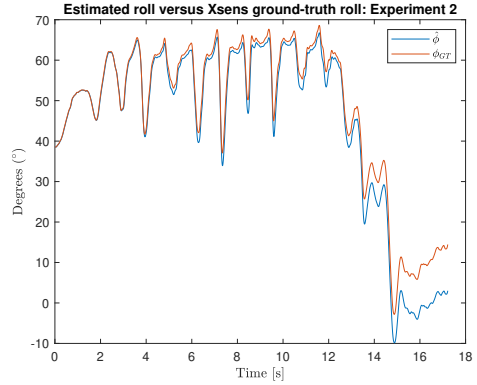
(a) Roll angle of the Xsens ground-truth versus the roll angle that is obtained via Algorithm 3 for the first experiment.



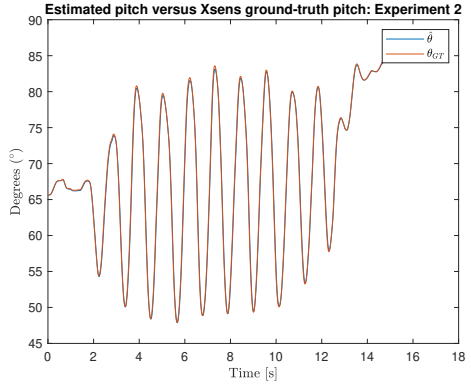
(b) Pitch angle of the Xsens ground-truth versus the pitch angle that is obtained via Algorithm 3 versus the first experiment.

Figure 4-4: Roll and pitch angles during experiment one, where the IMU is slowly rotated and a correct inclination estimate is obtained.

For the first experiment, an RMSE of 0.7028° and 0.7334° were obtained for the roll and pitch, respectively. This is a substantial decrease in RMSE values in comparison to the values that we found for algorithm 2. We immediately observe that the estimation is less noisy, which indicates better robustness against noise. For the second experiment, when the subject was asked to walk in a straight line, we found the following results shown in Figure 4-5.



(a) Roll angle of the Xsens ground-truth versus the roll angle that is obtained via algorithm 3



(b) Pitch angle of the Xsens ground-truth versus the pitch angle that is obtained via algorithm 3

Figure 4-5: Roll and pitch angle during experiment two, where the participant is asked to walk in a straight line while wearing an MTw Awinda on the left wrist.

For the second experiment, an RMSE of 0.5615° and 0.5226° were obtained for the roll and pitch, respectively. These values are lower than the RMSE values obtained in experiment one, where the sensor is only rotated. This was not expected, since we would expect a higher RMSE

when external accelerations are induced upon the accelerometer measurements. As mentioned before, it is important to note that using Xsens orientation estimates as a ground-truth may only serve as a tool to obtain a good initial insight in how one's algorithm performs, but can not be used to validate an algorithm thoroughly. As can be seen, both estimations of the roll and pitch are, unlike in Figure 3-4 able to converge towards the Xsens ground-truth. There are no problems regarding the violations of the assumed measurement model. Therefore, we can deem Algorithm 3, suitable as an extension for upper-limb prosthesis research and we can confidently state that when two sensors are placed upon both wrists, it is possible to obtain a very accurate estimation of the lower arm's inclination, which can tell us something about the characteristic movement patterns and the use of the prosthesis.

4-5 Activity Scoring System V.2.

Because of the additional gyroscope that is used, we can extend the Activity Scoring System that was created in Section 3-4. The tool from [16] and the tools created in Section 3-4 are limited regarding their analyses of what type of movement is performed while an activity is performed. With those methods, the acceleration, i.e., linear movement, is analyzed while ignoring all rotational movement. Here the extension for the Activity Scoring System is created where we include information of the rotational velocity from the arms. We introduce a similar type of vector magnitude (VM), as seen in Section 3-4. Here it is not required to subtract any number as seen in Eq. (3-20) because angular velocities are zero when the sensor lays still. The VM function for the gyroscope is given as

$$\mathrm{VM}_{\omega,t} = \|y_{\omega,t}\|_2 \,. \tag{4-14}$$

Here the VM indicates how much rotational velocity was performed during an activity. Similar to 3-4, the equation for the calculation of the VM does not take into account the sensor noise induced upon the measurement signal. Hence, Eq. (4-14) will always be non-zero. To determine whether no rotational movement occurs, we can look at the gyroscope measurements in all three directions to determine if the sensor lays still. Similar to Section 3-4, we will make us of an upper and lower bound created out of a chosen amount of standard deviations of the gyroscope. If we detect that the gyroscope noise falls between those bounds in all three directions, we set $\mathrm{VM}_{\omega,t} = 0$. We can also create a scoring system where we rank the VM. This scoring system also requires a high-intensity VM number to compare other VM. The scoring system then becomes

$$\mathrm{S}_{a,t} = 100 \cdot \frac{\mathrm{VM}_{\omega,t}}{\mathrm{VM}_{\omega,high}}. \tag{4-15}$$

Similar to Section 3-4, we can create epochs for the VM that are obtained via the gyroscope with epoch length l_e and sampling time T as

$$\mathrm{VM}_{\omega,e} = T \cdot \frac{1}{l_e} \left(\mathrm{VM}_{\omega,t} + \ldots + \mathrm{VM}_{\omega,t+\frac{l_e}{T}} \right), \tag{4-16}$$

and for the scoring system as

$$\mathrm{S}_{\omega,e} = T \cdot \frac{1}{l_e} \left(\mathrm{S}_{\omega,t} + \ldots + \mathrm{S}_{\omega,t+\frac{le}{T}} \right). \tag{4-17}$$

Together with these epochs, we can create a new method to show a percentage-wise contribution from the VM that show the contribution per arm regarding rotational movement as

$$\mathrm{Contribution}_{\mathrm{Dom}} = \left[\text{ round } \left(\frac{\mathrm{VM}_{D,\omega,e}}{\mathrm{VM}_{D,\omega,e} + \mathrm{VM}_{ND,\omega,e}} \cdot 100 \right) \right], \tag{4-18}$$

and for the non-dominant

$$\mathrm{Contribution}_{\mathrm{Non-Dom}} = \left[\text{ round } \left(\frac{\mathrm{VM}_{ND,\omega,e}}{\mathrm{VM}_{D,\omega,e} + \mathrm{VM}_{ND,\omega,e}} \cdot 100 \right) \right], \tag{4-19}$$

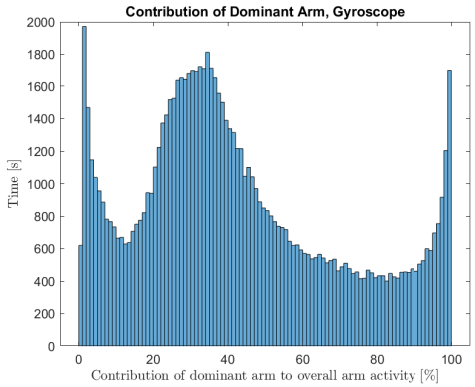
here $\mathrm{VM}_{D,\omega,e}$ is the value from epoch, e, of the VM that is created from rotational data for the dominant hand, and $\mathrm{VM}_{ND,\omega,e}$ for the non-dominant hand. This tool gives the prosthesis researcher an additional option to choose from and removes a certain part of the limitations currently exists.

4-6 Experimental Results of the Activity Scoring System V.2

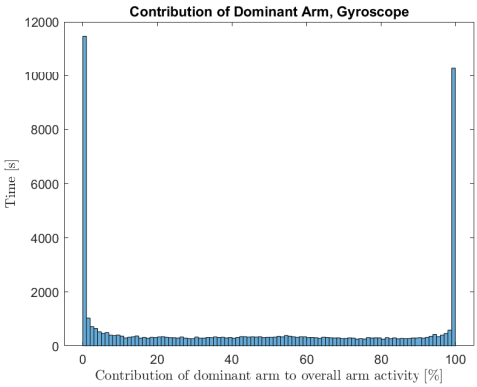
This section will discuss how the second scoring system performs when we use the same data-set that was introduced in Section 3-5. This data-set was collected with two AX6 monitors from Axivity, with the participants right hand being the dominant hand. For a detailed description of how the data-set is created, please refer to Section 3-5. From the manually extracted data to determine the standard deviation of the gyroscope, we obtained a value of $\sigma_\omega = 0.4909$ deg/s. This value was used for all three independent axis of the gyroscope to determine whether the VM is classified as zero. For the high intensity VM we found $\mathrm{VM}_{\omega,high} = 87$ deg/s. In our first experiment, we used a bound of $\pm 0.5\sigma_\omega$, and during our second experiment, a bound of $\pm 6\sigma_\omega$ was used. These bounds were chosen since these gave insightful results to discuss.

When the VM of both arms was equal to zero, we removed it from the data-set. Next, percentage bands from 0-100% with 1% increments were created. The time for each contribution band was added up and is displayed in seconds. For the first and second experiment, we obtained two histograms shown in Figure 4-6.

We observe similar results, as we saw in Section 3-5. When the bounds were chosen relatively small, we obtained a similar histogram as the ones presented in [16], which could indicate that the histogram is correct and that the bounds chosen for that situation are also correct. When the bounds' size was increased, we saw the same phenomena as to what happened when we increased the value of the bounds for the accelerometer; most bilateral movement is wrongly classified as zero.



(a) The histogram, which corresponds to the first experiment where we chose a standard deviation bound of $\pm 0.5\sigma_\omega$, we obtained a histogram which has a similar shape when compared to the histograms of [16].

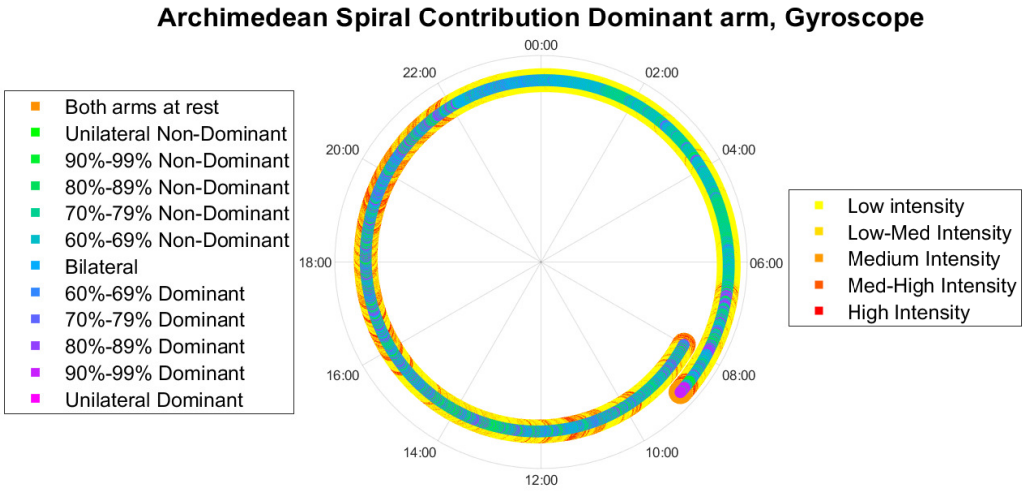


(b) The histogram, which corresponds to the second experiment where we chose a standard deviation bound of $\pm 6\sigma_\omega$, we obtained a histogram which does not have a similar shape when compared to the histograms of [16].

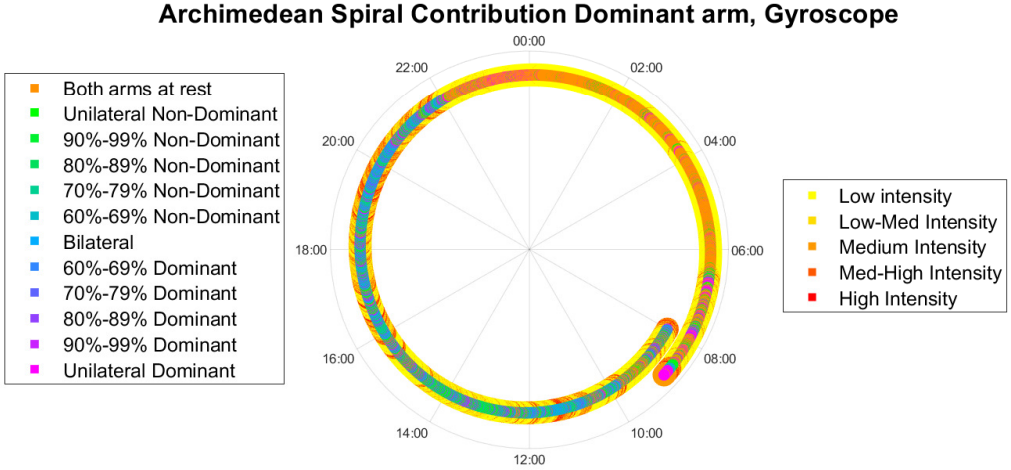
Figure 4-6: The histograms that correspond to the first and second experiment when the VM are created from gyroscopic data. Similar to the histograms that are created from the accelerometer measurements, we observe that a lower bound allows us to more accurately capture bilateral movement, while increasing the bound results in an inaccurate representation of the contribution of the arm to the overall activity.

Again, we create two spiral plots that overlap each other such that we can observe the type of movement and its intensity in one figure. The spiral plots for the two experiments are shown in Figure 4-7.

In Figure 4-7, we observe that a correct spiral plot is given for the second experiment. This is due to the same reason that we observed in Section 3-5, which is the increase in the standard deviation bound. This leads to a more accurate way of classifying the moment when both arms are at rest. Furthermore, it should be noted that when we experimented with values between $\pm 0.5\sigma_\omega$ and $\pm 6\sigma_\omega$, the histogram, as well as the spiral plots, were incorrect. This leads us to believe that when this happens, the classification of low-intensity bilateral movement and no movement is done incorrectly. From the experimental results in this section and the ones in Section 3-5, we can conclude that there is a trade-off between the classification of movement. By increasing the bound's value, we can classify the moment when both arms at rest more accurately, but lose the capability to classify bilateral movement correctly. This results in an increase in the wrong classification of unilateral movement and both arms at rest. By decreasing the value of the bounds, the opposite occurs; when the value of the bounds is decreased, we increase the accuracy in classifying bilateral movement but lose accuracy in classifying both arms at rest. Therefore, to overcome the problem regarding the bounds, future studies are required. Although there are some problems regarding the choice of the bounds, the scoring system designed to indicate the intensity of movement works. An idea might be to incorporate the activity counts (AC) from ActiGraph and create histograms and spiral plots with those measurements.



(a) The spiral plot that is created after the first experiment which handles a standard deviation bound of $\pm 0.5\sigma_\omega$. We observe that during the night the wrong classifications are made, which may lead to believe that the participant is performing constant movement during a 24 hour period.



(b) The spiral plot that is created after the second experiment which handles a higher standard deviation bound than experiment one, namely $\pm 6\sigma_\omega$. A correct representation of arm movement is created together with a correct display of the intensity of the movement.

Figure 4-7: The two spiral plots that correspond to the first and second experiment, in which the first experiment handles a bound of $\pm 0.5\sigma_\omega$ and the second $\pm 6\sigma_\omega$. Only the second plot is correct due to the larger value of the bounds.

Chapter 5

Prosthesis Analysis with Two Inertial Measurement Units Per Arm

This chapter shows how we can obtain an accurate relative sensor orientation estimation when two sensors are placed on adjacent segments [61]. These relative sensor orientations are clinically relevant since they show how the segments that sensors are placed on move with respect from each other. From large quantities of data, patterns may be discovered that might be insightful for the prosthesis researcher. We develop a computationally, efficient, and robust complementary filter that uses constraints related to the human body and its kinematics. The complementary filter opens up possibilities to process enormous data-sets, and due to the use of joint kinematic constraints, the use of a magnetometer is no longer required. After the mathematical explanation behind the algorithm, we numerically test and show different methods for estimating the lengths from the joint center (jc) to the Inertial Measurement Unit (IMU). After this, several numerical simulations indicate the robustness of the complementary filter where the filter will be tested against several types of commonly occurring disturbances, including outliers and Soft-Tissue-Artifacts (STA). We will also show the effects of the bias estimation algorithm from [31] that was presented earlier in Section 2-2-5 and the effects of having a varying value for β when no movement or high levels of STA occur. At last, we compare our algorithm versus the Multiplicative Extended Kalman Filter (MEKF) approach proposed in [61] in the experimental results.

5-1 Biomechanical Model Choice

As mentioned before in Section 2-3-1, the parameterization used for the kinematic chain model is mostly chosen to be minimal; these minimal orientation parameterizations suffer from singularities. Another disadvantage is that the calibration parameters, e.g., the lengths of the different segments and the poses of the sensors relative to the model, are assumed to be fixed and known. The latter will not be the case for this research project. The lengths of the body segments are user-specific parameters that need to be determined only once,

but the sensor poses and, in particular, their orientation relative to the segments must be re-calibrated every time the sensor network is placed on the subjects body [9, 8]. For the free segment model, a non-minimal parameterization is chosen to represent each segment with an orientation and position with respect to the global frame. Due to the non-minimal form of the parameterization and the stochastic constraints, the computational time will be higher than when a kinematic chain is used. This is the biggest drawback of utilizing this method. Nevertheless, this method does not require the calibration parameters to be known or estimated. Therefore, this method can be deemed sub-optimal for this research project.

5-2 Relative Sensor Orientation Estimation with a Complementary Filter

Here we will present a novel method to estimate the relative sensor orientation. Below a complementary filter is shown that uses two gyroscopes and two accelerometers, e.g., two Inertial Measurement Units (IMUs). By using the joint kinematic constraints induced by the human body itself, we can obtain a relative sensor orientation estimate that will not drift. To create such an estimate we require an accelerometer measurement model that assumes that there is movement; hence we choose [61]

$$y_{a,t}^{S_i} = R(q_t^{S_iG})\left(a^G - g^G\right) + e_{a,t}^{S_i}, \tag{5-1}$$

here, each segment consists of an inertial sensor with its own sensor coordinate frame, which we denote as S_i, with index i (where $i = 1, 2$), as can be seen in Figure 5-1.

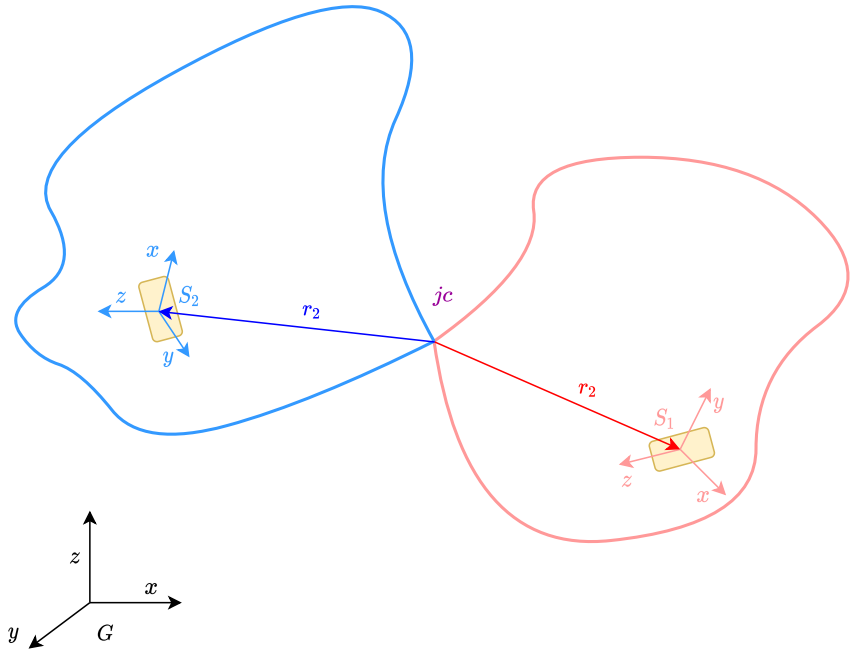


Figure 5-1: Two inertial sensor placed on adjacent segments, that are connected at a jc, where each IMU is placed at a distance r_i from the jc.

Furthermore, we assume that $y_{a,t}^{S_i}$ has unit norm and $g^G = [0 \quad 0 \quad 1]$. For the gyroscope's measurement model we choose

$$y_{\omega,t}^{S_i} = \omega_t^{S_i} + e_{\omega,t}^{S_i}. \tag{5-2}$$

For now, no bias on the gyroscope is included in the measurement model. Later in this chapter, we will treat the case when a gyroscope bias is added to the measurement model. We update both orientations $q_t^{GS_1}$ and $q_t^{GS_2}$ together, which allows us to compensate for drift in the relative sensor orientation $q_t^{S_1S_2}$ [61]. As mentioned before in Section 2-3-2. this can be done by utilizing common information present in the gyroscope and accelerometer measurements of two adjacent IMUs. We know that the jc, should have a common expression for its acceleration $a_{\text{jc},t}^G$, which is shown in Figure 5-2.

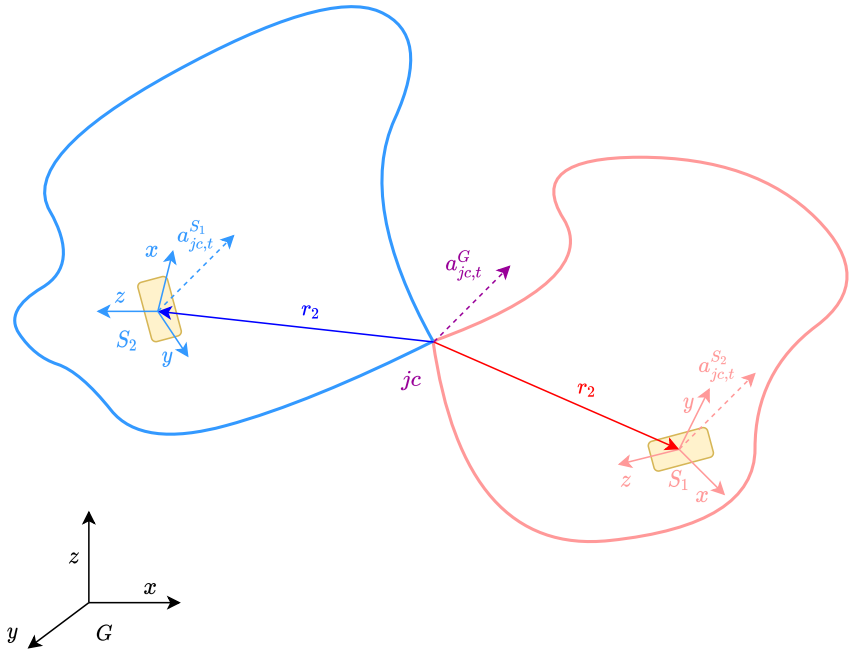


Figure 5-2: The joint accelerations in the first and second sensor frame, shown with the joint acceleration in the global frame. The joint accelerations from the first and second sensor frame can be rotated to obtain the joint acceleration in the global frame.

Due to the measurement noises and model inaccuracies in Eq. (5-1) and Eq. (5-2), errors arise in the rotation from the first and second sensor frame to the global frame. Hence, the difference of the acceleration in the jc obtained by rotation will have an induced error as

$$R(q_t^{GS_1})a_{\text{jc},t}^{S_1} = R(q_t^{GS_2})a_{\text{jc},t}^{S_2} + e_{\text{link},t}. \tag{5-3}$$

Here, $e_{\text{link},t} \sim \mathcal{N}(0, \Sigma_{\text{link}})$ and the jc acceleration in both the first and second sensor frame are calculated by utilizing the accelerometer measurement $y_{a,t}^{S_i}$ at a distance from the jc to the IMU by using $\mathcal{C}_t^{S_i}$ as

$$\begin{aligned} a_{\text{jc},t}^{S_i} &= y_{a,t}^{S_i} - \mathcal{C}_t^{S_i} r_i^{S_i}, \\ \mathcal{C}_t^{S_i} &= \left[y_{\omega,t}^{S_i}\times\right]^2 + \left[\dot{y}_{\omega,t}^{S_i}\times\right]. \end{aligned} \tag{5-4}$$

The $\mathcal{C}_t^{S_i}$ contains the time derivative of the angular velocity, i.e., the angular acceleration. The vectors r_1 and r_2 display the distances from the jc to the first and second IMU respectively.. As discussed in Section 2-2-4, the complementary filter from [24, 31, 32], use the gyroscope measurement to create an estimate of the orientation, which is accurate on a short time scale but is prone to drift. Therefore, they choose to use an orientation estimate obtained from the accelerometer and magnetometer, which is less accurate but drift-free. These two estimates, which have complementary properties, are used within a complementary filter. We will also use the orientation estimate of the gyroscope $q_{omega,t}^{GS_i}$. However, this entails that the relative orientation estimate $q_{\omega,t}^{S_1S_2}$ from the two different gyroscopes is also accurate on a short time scale, but will start to drift over a longer time horizon. To overcome drift in the relative sensor orientation, we can create an orientation estimate that contains information about the joint kinematics, which we denote as $q_{jc}^{GS_i}$. This relative orientation estimate, denoted by $q_{jc,t}^{S_1S_2}$, will generate a less accurate estimation due to it being more noisy, but will not drift, unlike $q_{\omega,t}^{S_1S_2}$. The accuracy property of the gyroscope estimate and the non-drifting property of the joint kinematics estimate can be combined using a complementary filter, in which $q_{jc}^{GS_i}$ is low-pass filtered and $q_{\omega}^{GS_i}$ is high-pass filtered. This will result in the following filtered orientation $\hat{q}^{GS_i}$ as

$$\begin{aligned} \hat{q}_t^{GS_i} &= (1-\gamma_t)\, q_{jc,t}^{GS_i} + \gamma_t q_{\omega,t}^{GS_i}, \\ &= (1-\gamma_t)\, q_{jc,t}^{GS_i} + \gamma_t \left(\hat{q}_{t-1}^{GS_i} + T\omega_{q,t}^{S_i}\right), \end{aligned} \tag{5-5}$$

where γ_t and $(1-\gamma_t)$ are weights applied to each orientation, with $0 \leq \gamma_t \leq 1$, T denotes the sampling time, and $\omega_{q,t}^{S_i}$ represents the angular velocity expressed in terms of a quaternion of the i-th sensor. Observing Eq. (5-5), it should be noted that $\hat{q}_t^{GS_i}$ does not represent a valid rotation since the quaternion is no longer of unit length. Later on, it will become clear that the deviation in length of this quaternion is small due to the high-frequency rates of the sensors. By utilizing an Euler discretization and assuming that there is a constant angular velocity between two time-samples, the dynamics of the orientation can be expressed in terms of the sampling time and the angular velocity as [23]

$$\begin{aligned} q_t^{GS_i} &= q_{t-1}^{GS_i} \odot \exp_{\text{q}}\left(\frac{T}{2} y_{\omega,t}^{S_i}\right), \\ &\approx q_{t-1}^{GS_i} + \frac{T}{2} S\left(q_{t-1}^{GS_i}\right) y_{\omega,t}^{S_i}. \end{aligned} \tag{5-6}$$

Where similar to [24], $S(q)$ is defined as

$$S(q) = \begin{bmatrix} -q_v \\ q_0 \mathcal{I}_3 - [q_v\times] \end{bmatrix}, \tag{5-7}$$

and the angular velocity is expressed in terms of a quaternion as

$$\omega_{q,t}^{S_i} = \frac{1}{2} S\left(\hat{q}_{t-1}^{GS_i}\right) y_{\omega,t}^{S_i}. \tag{5-8}$$

With the help of Eq. (5-3), the estimation of the orientation from the joint kinematics can be obtained by minimizing the following objective function as

$$\min_{q_t^{GS_i}} V\left(q_t^{GS_i}\right) = \min_{q_t^{GS_i}} \frac{1}{2} \left\| R(q_t^{GS_1}) a_{\mathrm{jc},t}^{S_1} - R(q_t^{GS_2}) a_{\mathrm{jc},t}^{S_2} \right\|_2^2. \tag{5-9}$$

The objective function of Eq. (5-9), can also be rewritten, with $\eta_t = \begin{bmatrix} \eta_{S_1,t}^\top & \eta_{S_2,t}^\top \end{bmatrix}^\top$, which is inspired by [24]. To do so, we use Eq. (5-3) but write the orientation in terms of a linearization point and an associated deviation instead of using unit-quaternions, which uses the approximations that were presented in section 2-2-4, in Eq. (2-29) and Eq. (2-30) as

$$\begin{aligned} \min_{\eta_t} V(\eta_t) &= \min_{\eta_t} \frac{1}{2} \left\| R\left(\tilde{q}_t^{GS_1}\right) \exp_{\mathrm{R}}(\eta_{S_1,t}) a_{jc,t}^{S_1} - R\left(q_t^{GS_2}\right) \exp_{\mathrm{R}}(\eta_{S_2,t}) a_{jc,t}^{S_2} \right\|_2^2, \\ &= \min_{\eta_t} \frac{1}{2} \left\| R\left(\tilde{q}_t^{GS_1}\right) (\mathcal{I}_3 + [\eta_{S_1,t}\times]) a_{jc,t}^{S_1} - R\left(q_t^{GS_2}\right) (\mathcal{I}_3 + [\eta_{S_2,t}\times]) a_{jc,t}^{S_2} \right\|_2^2. \end{aligned} \tag{5-10}$$

It should be noted that these approximations may only be used for when η_t is small. The reason for choosing this same method as [24] is to create the possibility of optimizing over an orientation deviation in terms of a rotation vector, rather than using a unit-quaternion. Therefore we avoid issues regarding quaternion normalization, and as an additional consequence of using orientation deviation states instead of a unit-quaternion, the length of the optimization variables is reduced from four to three. This will help us to create an algorithm, which is computationally efficient. Like [24], we solve the objective function of Eq. (5-10) in a single gradient descent iteration, instead of solving the problem for each time step. The benefit of doing a single gradient descent iteration is the decrease in computational time, and because of the high sampling frequency of the sensors, the corrections that need to be made are typically minor, and the estimates will converge over time. Linearising $V(\eta_t)$ from Eq. (5-10) around $\tilde{q}_t^{GS_i} = \hat{q}_{t-1}^{GS_i}$ and $\eta_t = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$ using Eq. (2-29) and Eq. (2-30), the gradient descent steps is given by

$$\hat{\eta}_t = -\mu_t \nabla V(\eta_t), \tag{5-11a}$$

$$\begin{aligned} \nabla V(\eta_t) = \Bigg(&-R\left(\hat{q}_{t-1}^{GS_1}\right) \begin{bmatrix} \left[a_{jc,t}^{S_1}\times\right] & 0_3 \end{bmatrix} + R\left(\hat{q}_{t-1}^{GS_2}\right) \begin{bmatrix} 0_3 & \left[a_{jc,t}^{S_2}\times\right] \end{bmatrix} \Bigg)^\top \\ &\left(R\left(\hat{q}_{t-1}^{GS_1}\right) a_{\mathrm{jc},t}^{S_1} - R\left(\hat{q}_{t-1}^{GS_2}\right) a_{\mathrm{jc},t}^{S_2} \right). \end{aligned} \tag{5-11b}$$

here μ_t is the gradient descent step length, and 0_3 is a 3×3 matrix consisting out of zeros. The derivative found in Eq. (5-11b) can be computed similarly to the computations seen in Section 4-2 as

$$\nabla V(\eta_t) = \frac{\partial V(\eta_t)}{\partial \eta_t} = \left(\frac{\partial e_t}{\partial \eta_t}\right)^\top e_t. \tag{5-12}$$

Here $e_t = R\left(\tilde{q}_t^{GS_1}\right)\left(\mathcal{I}_3 + [\eta_{S_1,t}\times]\right) a_{jc,t}^{S_1} - R\left(q_t^{GS_2}\right)\left(\mathcal{I}_3 + [\eta_{S_2,t}\times]\right) a_{jc,t}^{S_2}$. Again we choose to first create the derivative of e_t with respect to η_t, and transpose it at a later moment. To do so we have to asses the following

$$
\begin{aligned}
e =& R\left(\tilde{q}_t^{GS_1}\right)\left(\mathcal{I}_3 + [\eta_{S_1,t}\times]\right) a_{jc,t}^{S_1} - R\left(q_t^{GS_2}\right)\left(\mathcal{I}_3 + [\eta_{S_2,t}\times]\right) a_{jc,t}^{S_2},\\
=& R\left(\tilde{q}_t^{GS_1}\right)\left(\mathcal{I}_3 + [\eta_{S_1,t}\times]\right) a_{jc,t}^{S_1} - R\left(q_t^{GS_2}\right)\left(\mathcal{I}_3 + [\eta_{S_2,t}\times]\right) a_{jc,t}^{S_2},\\
=& R\left(\tilde{q}_t^{GS_1}\right)\begin{bmatrix} 1 & -\eta_{3,S_1,t} & \eta_{2,S_1,t} \\ \eta_{3,S_1,t} & 1 & -\eta_{1,S_1,t} \\ -\eta_{2,S_1,t} & \eta_{1,S_1,t} & 1 \end{bmatrix}\begin{bmatrix} a_{1,jc,t}^{S_1} \\ a_{2,jc,t}^{S_1} \\ a_{3,jc,t}^{S_1} \end{bmatrix}\\
& - R\left(\tilde{q}_t^{GS_2}\right)\begin{bmatrix} 1 & -\eta_{3,S_2,t} & \eta_{2,S_2,t} \\ \eta_{3,S_2,t} & 1 & -\eta_{1,S_2,t} \\ -\eta_{2,S_2,t} & \eta_{1,S_2,t} & 1 \end{bmatrix}\begin{bmatrix} a_{1,jc,t}^{S_2} \\ a_{2,jc,t}^{S_2} \\ a_{3,jc,t}^{S_2} \end{bmatrix},\\
=& R\left(\tilde{q}_t^{GS_1}\right)\begin{bmatrix} a_{1,jc,t}^{S_1} - \eta_{3,S_1,t}a_{2,jc,t}^{S_1} + \eta_{2,S_1,t}a_{3,jc,t}^{S_1} \\ \eta_{3,S_1,t}a_{1,jc,t}^{S_1} + a_{2,jc,t}^{S_1} - \eta_{1,S_1,t}a_{3,jc,t}^{S_1} \\ -\eta_{2,S_1,t}a_{1,jc,t}^{S_1} + \eta_{1,S_1,t}a_{2,jc,t}^{S_1} + a_{3,jc,t}^{S_1} \end{bmatrix}\\
& - R\left(\tilde{q}_t^{GS_2}\right)\begin{bmatrix} a_{1,jc,t}^{S_2} - \eta_{3,S_2,t}a_{2,jc,t}^{S_2} + \eta_{2,S_2,t}a_{3,jc,t}^{S_2} \\ \eta_{3,S_2,t}a_{1,jc,t}^{S_2} + a_{2,jc,t}^{S_2} - \eta_{1,S_1,t}a_{3,jc,t}^{S_1} \\ -\eta_{2,S_2,t}a_{1,jc,t}^{S_2} + \eta_{1,S_2,t}a_{2,jc,t}^{S_2} + a_{3,jc,t}^{S_2} \end{bmatrix}.
\end{aligned}
$$

With this, we can create the derivative as

$$
\begin{aligned}
\frac{\partial e}{\partial \eta} =& R\left(\tilde{q}_t^{GS_1}\right)\begin{bmatrix} 0 & a_{3,jc,t}^{S_1} & -a_{2,jc,t}^{S_1} & 0 & 0 & 0 \\ -a_{3,jc,t}^{S_1} & 0 & a_{1,jc,t}^{S_1} & 0 & 0 & 0 \\ a_{2,jc,t}^{S_1} & -a_{1,jc,t}^{S_1} & 0 & 0 & 0 & 0 \end{bmatrix}\\
& - R\left(\tilde{q}_t^{GS_2}\right)\begin{bmatrix} 0 & 0 & 0 & 0 & a_{3,jc,t}^{S_2} & -a_{2,jc,t}^{S_2} \\ 0 & 0 & 0 & -a_{3,jc,t}^{S_2} & 0 & a_{1,jc,t}^{S_2} \\ 0 & 0 & 0 & a_{2,jc,t}^{S_2} & -a_{1,jc,t}^{S_2} & 0 \end{bmatrix},\\
=& -R\left(\tilde{q}_t^{GS_1}\right)\begin{bmatrix}\left[a_{jc,t}^{S_1}\times\right] & 0_3\end{bmatrix} + R\left(\tilde{q}_t^{GS_2}\right)\begin{bmatrix}0_3 & \left[a_{jc,t}^{S_2}\times\right]\end{bmatrix}.
\end{aligned}
$$

If we now substitute the derivative of e into Eq. (5-12), then $\nabla V(\eta_t)$ becomes

$$
\begin{aligned}
\nabla V(\eta_t) =& \left(-R\left(\tilde{q}_t^{GS_1}\right)\begin{bmatrix}\left[a_{jc,t}^{S_1}\times\right] & 0_3\end{bmatrix} + R\left(\tilde{q}_t^{GS_2}\right)\begin{bmatrix}0_3 & \left[a_{jc,t}^{S_2}\times\right]\end{bmatrix}\right)^\top\\
& \left(R\left(\tilde{q}_t^{GS_1}\right)\left(\mathcal{I}_3 + [\eta_{S_1,t}\times]\right) a_{jc,t}^{S_1} - R\left(q_t^{GS_2}\right)\left(\mathcal{I}_3 + [\eta_{S_2,t}\times]\right) a_{jc,t}^{S_2}\right),\\
=& \left(-R\left(\tilde{q}_t^{GS_1}\right)\begin{bmatrix}\left[a_{jc,t}^{S_1}\times\right] & 0_3\end{bmatrix} + R\left(\tilde{q}_t^{GS_2}\right)\begin{bmatrix}0_3 & \left[a_{jc,t}^{S_2}\times\right]\end{bmatrix}\right)^\top\\
& \left(R\left(\tilde{q}_t^{GS_1}\right)\left(\mathcal{I}_3 + [0\times]\right) a_{jc,t}^{S_1} - R\left(q_t^{GS_2}\right)\left(\mathcal{I}_3 + [0\times]\right) a_{jc,t}^{S_2}\right),\\
=& \left(-R(\hat{q}_{t-1}^{GS_1})\begin{bmatrix}\left[a_{jc,t}^{S_1}\times\right] & 0_3\end{bmatrix} + R(\hat{q}_{t-1}^{GS_2})\begin{bmatrix}0_3 & \left[a_{jc,t}^{S_2}\times\right]\end{bmatrix}\right)^\top\\
& \left(R(\hat{q}_{t-1}^{GS_1})a_{\mathrm{jc},t}^{S_1} - R(\hat{q}_{t-1}^{GS_2})a_{\mathrm{jc},t}^{S_2}\right).
\end{aligned}
$$

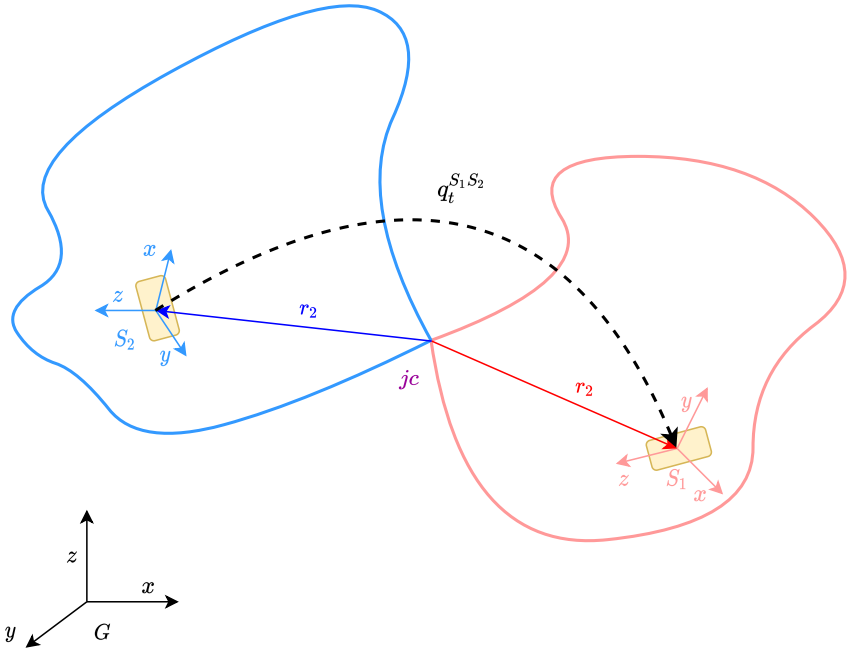


Figure 5-3: The relative sensor orientation between the first and second inertial sensor, that is independent of the global frame.

The estimate $\hat{\eta}_t$ can subsequently be used to compute $q_{jc,t}^{GS_i}$ from Eq. (5-5) as

$$q_{jc,t}^{GS_i} = q_{t-1}^{GS_i} \odot \exp_{\mathrm{q}}\left(\frac{1}{2}\hat{\eta}_{S_i,t}\right) \approx q_{t-1}^{GS_i} + \frac{1}{2}S\left(q_{t-1}^{GS_i}\right)\hat{\eta}_{S_i,t}. \tag{5-13}$$

The filtered orientation can be created by inserting (5-13) and (5-8) in (5-5) as

$$\hat{q}_t^{GS_i} = \hat{q}_{t-1}^{GS_i} + \frac{1}{2}S\left(\hat{q}_{t-1}^{GS_i}\right)\left(\gamma_t T y_{\omega,t}^{S_i} - \mu_t\left(1-\gamma_t\right)\nabla V\left(\eta_t\right)\right). \tag{5-14}$$

The only thing that is left is choosing the values for γ_t and μ_t. When the estimation mainly relies on the integration of the gyroscope measurement and uses joint kinematic constraints to correct for integration drift one typically chooses $\gamma_t \approx 1$ [31, 32]. Here, the scaling factor for the gradient descent direction, $\mu_t\left(1-\gamma_t\right)$, is chosen to be $\frac{\beta T}{\|\nabla V(\eta_t)\|}$. This is done to enhance the algorithm's robustness against measurement model imperfections. The choice of β depends on the amount of drift that is to be expected from the gyroscope measurement integration. When the noise is assumed to be $e_{\omega,t} \sim \mathcal{N}\left(0, \sigma_\omega^2\right)$. Integration of the gyroscope measurements in one dimension results in an integration drift distributed as $Te_{\omega,t} \sim \mathcal{N}\left(0, T^2\sigma_\omega^2\right)$ [24]. It is known that the gyroscope measurement is a three-dimensional vector, the standard deviation of the integration drift for the unit quaternion is then given by $\sqrt{3}\sigma_\omega T$ [24]. This is a good choice for βT. The resulting filter equation with $\hat{q}_t^{GS_i}$ is as

$$\hat{q}_t^{GS_i} \approx \hat{q}_{t-1}^{GS_i} + \frac{T}{2}S\left(\hat{q}_{t-1}^{GS_i}\right)\hat{\omega}_t^{S_i}, \tag{5-15}$$

with

$$\hat{\omega}_t^{S_i} = y_{\omega,t}^{S_i} - \beta \frac{\nabla V\left(\eta_{S_i,t}\right)}{\|\nabla V\left(\eta_{S_i,t}\right)\|}. \tag{5-16}$$

Then the relative orientation can be computed as

$$\hat{q}_t^{S_1 S_2} = \left(\hat{q}_t^{GS_1}\right)^c \odot \hat{q}_t^{GS_2}, \tag{5-17}$$

where the superscript c denotes the unit-quaternion conjugate, as shown in Eq. (C-26). A visualization of relative sensor orientation that is independent of the global frame is depicted in Figure 5-3. The final algorithm for the orientation estimation by using a complementary filter is given in Algorithm 4.

Algorithm 4 Relative sensor orientation estimation using an Complementary Filter with orientation deviation states

Input: Gyroscope & normalized accelerometer measurements $\left\{y_{\omega,t}^{S_i}, y_{a,t}^{S_i}\right\}_{t=1}^{N}$, Initial Orientation $\hat{q}_1^{GS_i}$, distance vectors r_1 & r_2, tuning parameter β and sampling time T.
Output: Relative sensor orientation estimate $\hat{q}_t^{S_1 S_2}$.

1: **for** $t = 2, \ldots, N$ **do**

$$\nabla V\left(\eta_t\right) = \begin{pmatrix} -R\left(\hat{q}_{t-1}^{GS_1}\right)\begin{bmatrix}\left[a_{jc,t}^{S_1}\times\right] & 0_3\end{bmatrix} + R\left(\hat{q}_{t-1}^{GS_2}\right)\begin{bmatrix}0_3 & \left[a_{jc,t}^{S_2}\times\right]\end{bmatrix} \\ \left(R\left(\hat{q}_{t-1}^{GS_1}\right) a_{\mathrm{jc},t}^{S_1} - R\left(\hat{q}_{t-1}^{GS_2}\right) a_{\mathrm{jc},t}^{S_2}\right) \end{pmatrix}^{\top}, \tag{5-18}$$

with $y_{a,t}^{S_i}$ and $\hat{q}_{t-1}^{GS_i}$.

2: Based on $\nabla V(\eta_t)$, compute

$$\hat{\omega}_t^{S_i} = y_{\omega,t}^{S_i} - \beta \frac{\nabla V\left(\eta_{S_i,t}\right)}{\|\nabla V\left(\eta_{S_i,t}\right)\|}. \tag{5-19}$$

3: Obtain the updated orientation

$$\hat{q}_t^{GS_i} \approx \hat{q}_{t-1}^{GS_i} + \frac{T}{2} S\left(\hat{q}_{t-1}^{GS_i}\right) \hat{\omega}_t^{S_i}, \tag{5-20}$$

with $S\left(\hat{q}_{t-1}^{GS_i}\right)$ as defined in Eq. (5-7).

4: Calculate the relative sensor orientation with

$$\hat{q}_t^{S_1 S_2} = \left(\hat{q}_t^{GS_1}\right)^c \odot \hat{q}_t^{GS_2} \tag{5-21}$$

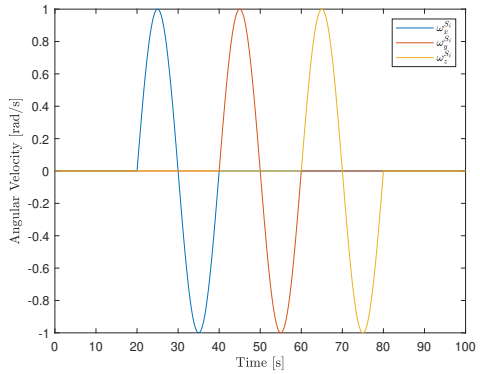
5: **end for**

5-3 Numerical Simulations of the Double Sensor Case

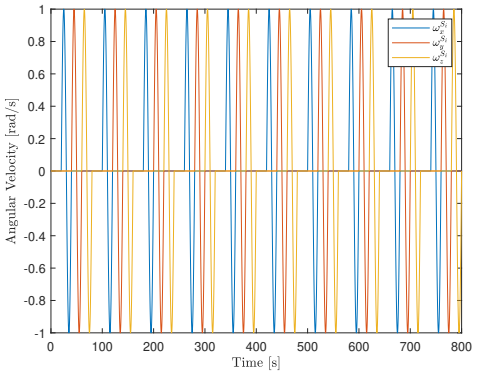
In this section, we will show different numerical simulations with the help of simulated datasets. We will treat several different topics, among which two methods for the estimation of

the distance to the IMU from the jc. After this, we will show how the complementary filter from Algorithm 4 can create estimates of the relative sensor orientation, and next, we will extend the gyroscope measurement model that was shown in Eq. (5-2) with a gyroscope bias and measure show the effects when we estimate that bias via the method that was presented in Section 2-2-5. At last, we will show the effects of using varying values for β during specific circumstances.

For all numerical simulations in this section, the basis of the simulations are all equal. The data-sets will contain the simulation of two distinct IMUs. Both of these IMUs have their angular velocities; to simulate the angular velocity measurement, we require a rotational speed. Here a the rotational speed of $\sin\left(\frac{\pi}{100T}t\right)$ rad/s will be assigned to IMU number one and $-\sin\left(\frac{\pi}{100T}t\right)$ rad/s to number two. This entails that both IMUs rotate in the opposite direction of each other. We chose these rotational speeds because the maximal rotational speed would then be one radiant per second, e.g., $\approx 57.3°$ per second, which is in the range of human-like movement. One cycle will take 800 samples, e.g., 80 seconds since the sampling frequency is equal to 10 HZ. During the first 200 samples, the simulated IMUs will not rotate; after not rotating for 200 samples, the IMUs will start with rotating around the x-axis with the rotational speed as mentioned above; after this, the IMUs will stop rotating in the x-direction and move on by rotating for 200 samples in the y-direction. The last 200 samples will be spent rotating around the z-axis; after this, the cycle will be repeated until 8000 samples are reached. In figure 5-4, one rotation cycle is depicted, as well as the angular velocity during the 8000 samples for the first IMU.



(a) The first rotational cycle with the first part of the second cycle, which show the rotational velocity of IMU number one.



(b) The full 800 seconds of simulated rotational speed, where a clear patterns shows in the rotational speed in the different directions.

Figure 5-4: The rotational speed for the first IMU, with a zoomed and full figure of the angular velocity.

.

From these angular velocities, we can easily create a derivative, which is the angular acceleration. For IMU number one this will be $\frac{\pi}{100T}\cos\left(\frac{\pi}{100T}t\right)$ rad/s^2 and $-\frac{\pi}{100T}\cos\left(\frac{\pi}{100T}t\right)$ rad/s^2 for the second IMU. The IMUs also require accelerometer measurements, these can be constructed from deconstructing Eq. (5-4). We first create the joint acceleration as a random signal between 10 and -10 m/s^2 for all three independent axis and rotate the joint acceleration signal to the first IMU sensor frame, from that signal, together with the angular velocity, angular acceleration and the distance to the jc we are able to create an acceleration signal for

IMU number one. This exact same method can be applied to the second IMU to obtain the acceleration signal for sensor two. The distances to the jc are chosen to be $r_1 = [1 \quad 0 \quad 0]^\top$ m and $r_2 = [-1 \quad 0 \quad 0]^\top$ m. We than proceed by adding simulated noise, as $e_{\omega,t} \sim \mathcal{N}(0, \Sigma_\omega)$ and $e_{a,t} \sim \mathcal{N}(0, \Sigma_a)$, where we set the standard deviation of the gyroscope noise as $\sigma_\omega = \frac{5\pi}{180}$ rad/s and for the normalized accelerometer as $\sigma_a = -\frac{g}{100}$.

5-3-1 The Estimation of the Length from the Joint Center to the Inertial Measurement Unit with a Sum of Squared Errors Objective Function

The motivation for the estimation of the distance vector is found in the joint acceleration calculation, which was shown in Section 5-2 in Eq. (5-4). Here we observe that the joint acceleration calculation requires the distance from the jc to the IMU. Since the distance vector is a user-specific parameter that may change per trial during real-world experimental setups, it is essential to create and understand how the distance vector can be obtained.

The length from the jc to the IMU can be estimated via the methods presented in [40], that were described in Section 2-3-2. In this section, the method that we will discuss is the one where the objective function is defined as a sum of squared errors. Here we use a Gauss-Newton algorithm to find $r = [r_1^\top \; r_2^\top]^\top$, which minimizes the cost function that is given as

$$\widehat{r} = \arg\min_{r} \sum_{t=1}^{N} e_{r,t}^2. \tag{5-22}$$

The Gauss-Newton algorithm uses the Jacobian as well as the Moore-Penrose-Pseudoinverse [59, 5], to solve this problem is as

$$r_s = r_{s-1} - \left(\left(\frac{\partial e_{r,s-1}}{\partial r_{s-1}} \right)^\top \left(\frac{\partial e_{r,s-1}}{\partial r_{s-1}} \right) \right)^{-1} \left(\frac{\partial e_{r,s-1}}{\partial r_{s-1}} \right)^\top e_{r,s-1}. \tag{5-23}$$

Here the values for r_s are updated for every Gauss-Newton iteration, where s is the number of iterations that can be chosen to run so that the Gauss-Newton algorithm updates the value of r, s times. For these numerical simulations, we have adopted the same error metric, as seen in [40]. In [40], a deliberate choice is made not to use Root Means Square Error (RMSE) values as an indication of the performance of the algorithm; instead, the performance of the estimators is evaluated in terms of the average Euclidean error and its standard deviation as

$$E_{\text{avg}}(\widehat{r}_i) = \frac{1}{M} \sum_{j=1}^{M} E_j(\widehat{r}_i), \tag{5-24}$$

with

$$E_j(\widehat{r}_i) = \|\widehat{r}_i(j) - r_i\|_2. \tag{5-25}$$

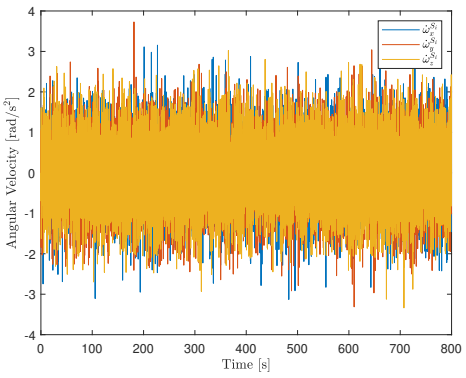
This is done because the RMSE values may be misapprehended as the average physical distance between the estimates with respect to the true jc [40].

We will run four different types of simulations were each simulation is an addition to the previous one. In all simulations the initial values for r_1 and r_2, are set to $[2 \quad 0 \quad 0]$ m and $[-2 \quad 0 \quad 0]$ m. As of now, in every simulation that requires the angular acceleration, we will use a five-point stencil approximation [55]. This allows us to create the first derivative of the angular velocity as

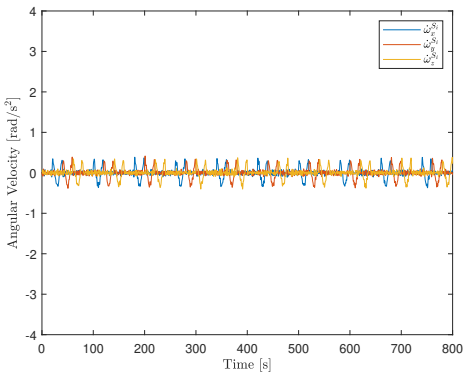
$$\dot{y}^{S_i}_{\omega,t} \approx \frac{-y^{S_i}_{\omega,t+2} + 8 \cdot y^{S_i}_{\omega,t+1} - 8 \cdot y^{S_i}_{\omega,t-1} + y^{S_i}_{\omega,t-2}}{12 \cdot T}. \tag{5-26}$$

For each simulation, we created 100 Monte Carlo simulations to perform the estimation of r, when we apply the Gauss-Newton algorithm, as shown in Eq. (5-23). A value of ten Gauss-Newton iterations is chosen to update the value for r before returning the final estimate. The decision for choosing ten iterations comes from [40], where it is stated that in almost every case, the estimates will have converged within ten Gauss-Newton iterations.

During the first simulation, we use the simulated data-set as described in the introduction of this Section, from this, we observe large values for our error metric, as shown in Table 5-1. Therefore, we can conclude that the estimation algorithm does not work correctly. The reason for this failure to get a correct, or even close to the correct estimate is that we have not filtered the approximation of the angular acceleration shown in Eq. (5-26). To do this, we have used MATLAB's built-in low-pass filter, in which we can specify the normalized passband frequency. Since we do not know the noise's exact frequency, we have set the normalized passband equal to 0.01 Hz. The low-pass filter uses a minimum-order filter with a stopband attenuation of 60 dB and compensates for the filter's delay. To compensate for the delay, the function appends to the input signal $\frac{O_f}{2}$ zeros, where O_f is the filter order. The function then filters the signal and removes the first $\frac{O_f}{2}$ samples of the output. Figure 5-5 shows the difference between the unfiltered and filtered estimate of the angular acceleration.



(a) Angular acceleration of IMU one, during the last Monte Carlo simulation, where the angular acceleration signal is undetectable due to the high amounts of noise present that corrupt the signal.



(b) The low-pass filtered angular acceleration signal of IMU one, during the last Monte Carlo simulation, where the signal is visible and can be used to obtain as an estimate of the angular acceleration.

Figure 5-5: The angular acceleration of the first IMU during the last Monte Carlo simulation. A clear increase in accuracy is obtained due the filtering of the high frequency noise.

.

Here it can be seen that the noise is correctly filtered out. For our second simulation, we

rerun our first simulate but include the low-pass filter on the angular acceleration signal. In Table 5-1, we observe a definite increase in accuracy with respect to when no low-pass filter is used, but still, the final results are not near the actual value of r, the reason for the value of r not being close enough to the true value is because of the induced gyroscopic noise. Therefore during the third simulation, we apply a rather unconventional method to solve this problem. We change the gyroscope signal by applying the same low-pass filter used to filter the approximation of the angular acceleration. By looking at Table 5-1, we can conclude that this method works and can generate an accurate estimate of the lengths from the jc to the IMU. Regarding the use of a low-pass on the gyroscope signal, it should be noted that the noise created on the gyroscope is of a rather large magnitude, but it was chosen this way since we wanted to resemble the simulated data created in [24]. The standard deviation of real gyroscopes noise, for example in an Xsens IMU, is presumably lower than the one that is chosen in this simulated data-set. For our last simulation with the objective function as a sum of squared errors, we added accelerometer outliers by replacing a certain percentage of the normalized accelerometer data with the outliers that are chosen sampled from a Gaussian distribution with an identity matrix as a covariance matrix. Again, we choose to replace 5% of the normalized accelerometer data and run the estimation algorithm for 100 Monte Carlo simulations. We observe that, minimizing over the sum of squared errors with a Gauss-Newton optimization is not very robust against outliers. This is reflected in the values obtained for the average Euclidean error, which are shown in Table 5-1.

		$\hat{r}_1$		$\hat{r}_2$	
Simulation		E_{avg}	σ_{avg}	E_{avg}	σ_{avg}
1	Known noise and variance	0.8709 m	0.0053 m	0.8722 m	0.0055 m
2	a low-pass filter on $\dot{y}_{\omega,t}$	0.0824 m	0.0080 m	0.0850 m	0.0075 m
3	a low-pass filter on $y_{\omega,t}$	0.0162 m	0.0059 m	0.0174 m	0.0062 m
4	5% outliers magnitude $\mathcal{N}(0, \mathcal{I})$	0.2450 m	0.1114 m	0.2597 m	0.1129 m

Table 5-1: Results of the numerical simulations for the estimation of the length from the jc to the IMU, when the objective function is a sum of squared error, which is optimized via a Gauss-Newton method. The characteristics of each simulation is an addition upon the previous one.

From these results, we can conclude that, due to the lack of robustness against accelerometer outliers, the distance vector's estimation via minimizing a sum of squared errors with a Gauss-Newton optimization method is not suitable for prosthesis research.

5-3-2 The Estimation of the Length from the Joint Center to the Inertial Measurement Unit with a Sum of Absolute Errors Objective Function

To counter the problems caused by outliers we can create an estimation algorithm that is robust against this. By defining the cost function in a different way we can ensure the

convergence of the parameter, while the parameter is subjected to outliers. The cost function is changed to

$$\widehat{r} = \arg\min_{r} \sum_{t=1}^{N} |e_{r,t}| \,. \tag{5-27}$$

As mentioned in Section 2-3-2, a known property of minimizing the sum of absolute error is robustness against outliers in the data-set [40]. This is because large residuals are weighted by their absolute value instead of by their squared value, as was seen in the sum of the squared error cost function. Since we try to optimize over a sum of absolute error, we can not use the Gauss-Newton optimization method anymore. Hence, we require a different gradient descent method. The optimization method that we choose is a backtracking line search in gradient descent [11]. With gradient descent we find a new point, for each iteration, that is closer to the local minimum of the function by moving the point in the direction of the descent as

$$r_s = r_s - \kappa \frac{\partial e_{r,s}}{\partial r_s} \tag{5-28}$$

By how much we move, the point is determined by the step size, also referred to as the learning rate, which in our case is denoted as κ. There are several ways of choosing this parameter, one of which is a line search. Line search is a way of evaluating the function for different values of κ and choosing the one that minimizes the function the most. Backtracking line search start by selecting a value for τ in the range from zero to one; this parameter determines how much we update the learning rate by each iteration. Conventionally the initial learning rate for this method is set to one. Next, we should evaluate the following inequality.

$$f\left(r - \kappa \frac{\partial e_{r,t}}{\partial r} f(r)\right) > f(r) - \frac{\kappa}{2} \left\| \frac{\partial e_{r,t}}{\partial r} \right\|^2 \tag{5-29}$$

where the function $f(r)$ is the cost function that we are trying to minimize is Eq. (5-27). Given that the inequality is true for the current κ, we update the learning rate by multiplying it by τ, such that it gets smaller by each iteration for which the inequality holds. If the inequality does not hold anymore for the current κ, we update the value of r with the learning rate κ, as was seen in Eq. (5-28). We have also incorporated our own stopping criteria for this algorithm, such that the algorithm would stop iterating if the decrease in the cost function would be smaller than 0.1 %.

We run five different types of simulations, where each simulation is 100 Monte Carlo simulations with $\tau = 0.8$ and where we have added outliers similar to Section 4-2. During the first simulation, we do not apply a low-pass filter on the gyroscope measurements, and we observe an apparent degradation regarding the accuracy of the estimation algorithm, as can be seen in Table 5-2. By adding the low-pass filter during the second simulation, we observe that the problems regarding the gyroscope noise disappear. From these results, we can conclude that whenever we expect or know that the gyroscope noise will be high, where high defines such a high variance value where we expect the estimate to suffer so much that it is not able to generate accurate parameter estimates anymore, a solution would be to also low-pass the gyroscope signal.

As mentioned before, the outliers that we have simulated try to emulate bumping into an object with the IMU attached to the arm, which generates a spike in the accelerometer signal.

However, this is not the only type of disturbance that frequently occurs when performing some evaluation that requires the IMUs to be worn on the human body. STA may occur due violations in the assumption that the sensor's position with respect to the jc is fixed at all times since the surface on which the sensors are placed can move to some degree. The STA can be simulated as an additive disturbance to the acceleration measurements as a linear combination of the angular acceleration measurements as [40]

$$d_a^{S_i} = H\dot{\omega}^{S_i} \tag{5-30}$$

Here the matrix H is a 3×3 matrix contains the STA model parameters. During real-world setups and testing, the elements of H will depend on multiple factors, i.e., sensor placement and the participant's physiology [40]. Therefore, we choose the element of H to be randomized as a zero-mean Gaussian distribution, where the $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad, which equals 0.057 m/rad. To give some insight into how this translates to human movement and the outcome of STA during an experiment, we show the STA signal of the first IMU in the simulated setup.

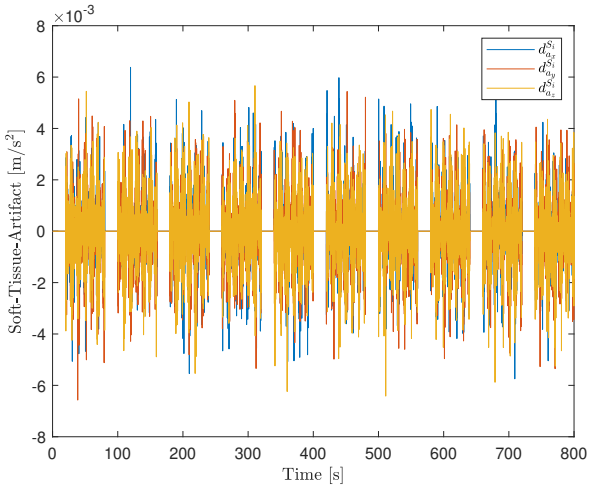


Figure 5-6: STA modeled as a randomized zero-mean Gaussian distribution multiplied with the angular acceleration. The STA levels are periodically, and sometimes will not occur. This is due to the gyroscope signal that also being periodically zero, which was shown in Figure 5-4.

In Figure 5-6, we see that the maximal value for the STA, with a standard deviation as mentioned above, is around $4 \cdot 10^{-3}$ m/s^2 if we apply an acceleration of this magnitude to a sensor, the displacement after 10^{-1} seconds would be roughly 40 μm, this is a relatively small value but will serve as a good approximation for low values of simulated STA.

As shown in Table 5-2, the algorithm can consistently reject the disturbances induced by low values of STA. By changing the standard deviation used in our H matrix, we can simulate both moderate and high STA levels. A good approximation for moderate levels of STA is chosen to be $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad, which roughly translate to a maximum displacement of 0.4 cm after 10^{-1} seconds. For high levels of STA we choose a STA standard deviation of $\sigma_{d_a} = \frac{18}{\pi}$ m/rad, which translates to a maximum displacement of 4 cm after 10^{-1} seconds. Table 5-2, shows that the estimation algorithm is robust for low and moderate values of STA, and is not fully capable of rejecting high levels of STA.

		$\hat{r}_1$		$\hat{r}_1$	
Simulation		E_{avg}	σ_{avg}	E_{avg}	σ_{avg}
1	Known noise, variance, 5% outliers magnitude $\mathcal{N}(0, \mathcal{I})$ a low-pass filter on $\dot{y}_{\omega,t}$	0.0876 m	0.0156 m	0.0882 m	0.0142 m
2	a low-pass filter on $y_{\omega,t}$	0.0205 m	0.0082 m	0.0214 m	0.0084 m
3	Low STA $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad	0.0221 m	0.0084 m	0.0228 m	0.0077 m
4	Moderate STA $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad	0.0241 m	0.0143 m	0.0244 m	0.0122 m
5	High STA $\sigma_{d_a} = \frac{18}{\pi}$ m/rad	0.0625 m	0.0633 m	0.0274 m	0.0304 m

Table 5-2: Results of the numerical simulations for the estimation of the length from the jc to the IMU, when the objective function is a sum of absolute error, which is optimized via a backtracking line search in gradient descent. Where the characteristics of each simulation is an addition upon the previous one.

We saw that the estimation via Gauss-Newton optimization works very well until outliers are introduced. Therefore we were required to develop another solution, which presented itself in the form of optimization over the sum of absolute errors. We chose to use a backtracking line search in gradient descent and saw that with this method, we were able to overcome the problem of outliers, and we can even conclude that the added low and moderate STA are of little influence regarding the accuracy of the estimate. For high levels of STA, we did start to notice some evident degradation regarding the performance, which is visible in Table 5-2. Therefore, this method can be used as a successful tool in prosthesis evaluation to estimate the distance from the jc to the IMU.

5-3-3 Numerical Simulations of the Relative Sensor Orientation Estimation with a Complementary Filter

In this section, we will perform simulations where we expose Algorithm 4 to noise, outliers, and different levels of STA. This is done because these types of disturbances are likely to occur during a real-world setup. To enable the testing of the algorithm, we, at first, assume that we have prior knowledge on the noise, the initial orientation, and the distance vectors r_1 and r_2. The initial orientation is chosen to be a $[1 \quad 0 \quad 0 \quad 0]^\top$ unit-quaternion and the exact location of r_1 and r_2, are $[1 \quad 0 \quad 0]^\top$ m and $[-1 \quad 0 \quad 0]^\top$ m. The error metric that is used is are RMSE values, since these give a clear overview about the magnitude of the error, these values will be supported with boxplots, and to indicate convergence from a wrong initial condition, we will show the mean error together with three standard deviations, to indicate the spread during convergence. We will cover several different simulations, where each time, we will add a different type of disturbance. Each simulation will consist out of 100 Monte

Carlo simulation, and it should also be noted that because of Eq. (5-26), we lose three-time steps, the first, second, and last one. We have added noise components on the gyroscope and the accelerometer during the first simulation, as described previously. In Table 5-3, we see that after performing 100 Monte Carlo simulations, the algorithm performs well under these circumstances. By replacing 5% of the normalized accelerometer measurements with outliers, which are zero-mean Gaussian distributed with a covariance equal to a 3×3 identity matrix, and performing the same simulation, we observe that Algorithm 4 has an outstanding capability to be robust against outliers. This will be an advantageous property in real-life prosthesis evaluation and research since accelerometer outliers will occur. Another significant disturbance to add is in the form of STA, here the STA are modeled as we saw previously in Eq. (5-30) and Figure 5-6. In simulation three, four, and five, we have added different types of STA values. After performing 100 Monte Carlo simulations for each simulation, we can conclude that for low and moderate values STA, the algorithm can correctly estimate the relative sensor orientation. However, if we increase the amount of STA, we observed a clear degradation in performance. High levels of STA can occur when a person performs an active movement where the flesh that the sensor is placed upon moves a lot. Later on, we will see that a forward lunge movement, when the sensor is placed upon the leg, is one of these movements.

Simulation		Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	Yaw ($\hat{\psi}$)	$\sigma_{\hat{\phi}}$	$\sigma_{\hat{\theta}}$	$\sigma_{\hat{\psi}}$
1	Known noise, variance and initial orientation	1.2968°	1.2922°	1.2927°	0.0171°	0.0152°	0.0144°
2	5% outliers magnitude $\mathcal{N}(0, \mathcal{I})$	1.4147°	1.4091°	1.4151°	0.0204°	0.0209°	0.0204°
3	Low STA $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad	1.4124°	1.4102°	1.4089°	0.0213°	0.0203°	0.0203°
4	Moderate STA $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad	1.5397°	1.5412°	1.5366°	0.0261°	0.0272°	0.0246°
5	High STA $\sigma_{d_a} = \frac{18}{\pi}$ m/rad	3.0062°	3.0056°	3.0065°	0.1168°	0.1244°	0.1091°

Table 5-3: Effects on the RMSE and standard deviation of the relative sensor orientation error. Algorithm 4 is robust and performs well, until a high level of STA is introduced during the last simulation.

To visualize the results from Table 5-3, we have created a boxplot of the five different simulations in Figure 5-7. Here, ϕ_i, θ_i and ψ_i with $i = 1, \ldots, 5$, denote the first until the fifth simulation for the relative roll, pitch and yaw respectively. From the boxplot, it can be seen that due to the higher levels of STA, the error in the three Euler angles increases. At last, we want to see if the algorithm presented in Algorithm 4 can converge while a wrong initial orientations is given, while the measurements are being subjected to noise, outliers, and moderate levels of STA. In Figure 5-8 we show the estimation error for the first 200 samples of 100 Monte Carlo simulations with a fixed wrong initial orientation as $[81.87° \quad -19.47° \quad 135.00°]$

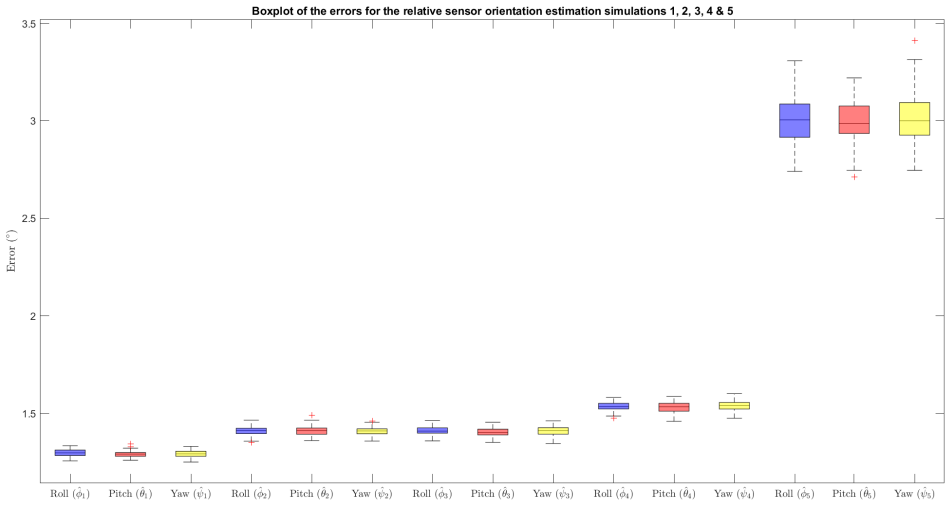


Figure 5-7: Boxplot of the five simulations for the relative sensor orientation, here the effects of increasing the STA levels is graphically made clear. The most important finding is the increase in error due to the increase in STA levels.

for sensor one and $[135.00^\circ \quad 19.47^\circ \quad 98.13^\circ]$ for sensor two. Here the thick dark-colored lines represent the mean of the relative roll, pitch, and yaw, and the shaded lighter-colored areas represent three standard deviations to capture 99.73% of the data created during this simulation. We observe a clear convergence towards zero, despite the moderate level of STA and outliers induced upon the accelerometer signal. From this, we can conclude that the Algorithm 4, is robust against noise, outliers, moderate values of STA, and can converge from a wrong initial orientation.

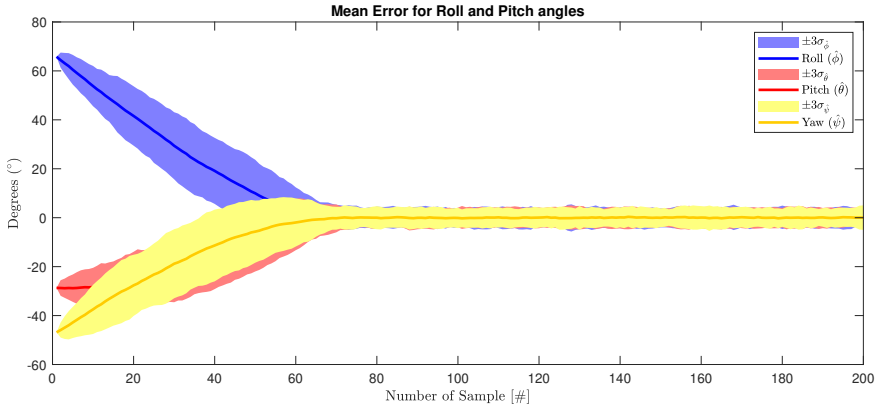


Figure 5-8: Estimation error for the first 200 samples of 100 Monte Carlo simulations with a fixed wrong initial orientation as $[81.87^\circ \quad -19.47^\circ \quad 135.00^\circ]$ for sensor one and $[135.00^\circ \quad 19.47^\circ \quad 98.13^\circ]$ for sensor two. Additionally, 5% simulated outliers and moderate levels of STA are induced upon the accelerometer signal.

5-3-4 Numerical Simulations for the Bias Estimation during Relative Sensor Orientation Estimation with a Complementary Filter

In Section 2-2-5, we described how in [31] a tool was created that can be incorporated to increase the accuracy of the estimation algorithm by estimating and removing the gyroscope bias. In this section, we will test this method. We will make use of six different simulations where every two simulations correspond in pairs which each other. In all six simulation we will add a constant gyroscopic bias of $[0.02 \quad -0.04 \quad 0.06]$ rad/s for sensor one and $[-0.08 \quad 0.07 \quad 0.03]$ rad/s for sensor two. During our first pair of simulations, we will add noise, outliers by replacing 5% of the normalized accelerometer measurements, and at the same time, we will also add low values of STA as described in earlier simulations. For our second pair of simulation, we will increase the values of the STA to moderate, and for the third pair of simulations, the levels of STA are set to high. In Table 5-4, the results for the relative sensor orientation are found when no bias estimation is applied, and when the bias estimation is applied to estimate the bias and remove it. In all cases, we see a clear difference between the simulations that have a bias estimation algorithm and the ones that do not. The simulations that did not enable the bias estimation algorithm from Section 2-2-5 have a clear degradation in performance, while the ones that do enable the bias estimation can generate accurate estimates of the relative sensor orientation.

Simulation		Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	Yaw ($\hat{\psi}$)	$\sigma_{\hat{\phi}}$	$\sigma_{\hat{\theta}}$	$\sigma_{\hat{\psi}}$
1	Low STA $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad No bias estimation	2.7816°	3.0213°	2.2308°	0.0832°	0.1012°	0.0674°
2	Low STA $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad With bias estimation	1.4643°	1.4692°	1.4606°	0.0223°	0.0240°	0.0203°
3	Moderate STA $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad No bias estimation	3.1000°	3.3864°	2.4617°	0.1032°	0.1175°	0.0813°
4	Moderate STA $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad With bias estimation	1.5907°	1.5952°	1.5861°	0.0259°	0.0265°	0.0300°
5	High STA $\sigma_{d_a} = \frac{18}{\pi}$ m/rad No bias estimation	8.0458°	9.1270°	5.6161°	0.2688°	0.3007°	0.1914°
6	High STA $\sigma_{d_a} = \frac{18}{\pi}$ m/rad With bias estimation	3.2322°	3.2381°	3.2387°	0.1213°	0.1197°	0.1268°

Table 5-4: Results of the numerical simulations for the estimation of the relative sensor orientation when the gyroscope measurements are corrupted by a constant gyroscopic bias of $[0.02 \quad -0.04 \quad 0.06]$ rad/s for sensor one and $[-0.08 \quad 0.07 \quad 0.03]$ rad/s for sensor two. Per simulation we enable or disable the bias estimation algorithm from [31].

In Table 5-5, the results regarding the accuracy of the bias estimate are given. We use the RMSE values as our error metric, and see that the bias estimate is accurate during the

Simulation		$\hat{b}_{\omega,1,x}$	$\hat{b}_{\omega,1,y}$	$\hat{b}_{\omega,1,z}$
2	Low STA $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad	0.0095 rad/s	0.0075 rad/s	0.0121 rad/s
4	Moderate STA $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad	0.0094 rad/s	0.0077 rad/s	0.0122 rad/s
6	High STA $\sigma_{d_a} = \frac{18}{\pi}$ m/rad	0.0115 rad/s	0.0094 rad/s	0.0143 rad/s

Simulation		$\hat{b}_{\omega,2,x}$	$\hat{b}_{\omega,2,y}$	$\hat{b}_{\omega,2,z}$
2	Low STA $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad	0.0125 rad/s	0.0096 rad/s	0.0107 rad/s
4	Moderate STA $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad	0.0126 rad/s	0.0097 rad/s	0.0108 rad/s
6	High STA $\sigma_{d_a} = \frac{18}{\pi}$ m/rad	0.0144 rad/s	0.0116 rad/s	0.0129 rad/s

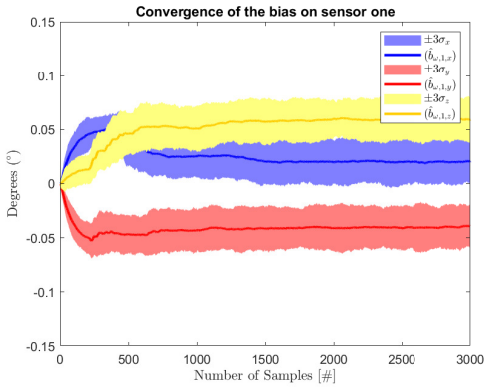
Table 5-5: RMSE values obtained via the the numerical simulations for the estimation of the bias estimation via [31]. The bias estimation method performs well until high levels of STA are introduced, when this happens the bias estimate becomes inaccurate.

second and fourth simulation, but loses accuracy when high values of STA are induced upon the accelerometer measurements. From this, we can conclude that the bias estimate could be of help during experimental trials, whenever we except low or moderate values of STA.

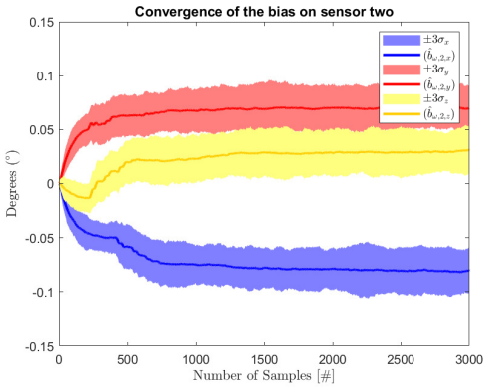
To give more insight into the convergence of the bias estimate, we created Figure 5-9, which shows the mean of the bias estimate for the first and second sensor, accompanied by three standard deviations to show the spread during the second simulation. If we compare the rate of convergence of the bias estimate with the rate of convergence of the relative sensor orientation estimate from Figure 5-8, we observe that the bias estimation algorithm requires approximately twice as much time to converge to a correct estimate.

5-3-5 Numerical Simulations for the use of a Varying β Value

This section will show the effects of having a dynamic value for our β parameter. As mentioned before, the value for β depends on the amount of drift expected from the integration of the gyroscope noise [24]. For our proposed algorithm, a varying beta could be beneficial in two cases; when we except no movement since the use of joint kinematic constraints requires the movement of the sensors, and when we expect high values of STA at certain time intervals. To create a simulated data-set for the first option, we remove the angular velocity and accelerations in the first 200 samples of every movement cycle, as was previously depicted in Figure 5-4a, and thus simulate that there is no movement of the sensor taking place. During this simulation, no outliers or STA are induced upon the accelerometer measurements, since in the real world, this would also not happen when no movement occurs. During the first simulation we only apply a static parameter value of $\beta = \sqrt{3}\sigma_\omega$, and during the second simulation, we apply a much lower value of β, namely $\beta = 0.01\sqrt{3}\sigma_\omega$ during the moments that there is no movement, and the regular $\beta = \sqrt{3}\sigma_\omega$ during the moments where we know



(a) Convergence of the bias estimate, where the constant where the constant gyroscopic bias is $[0.02 \quad -0.04 \quad 0.06]$ rad/s.

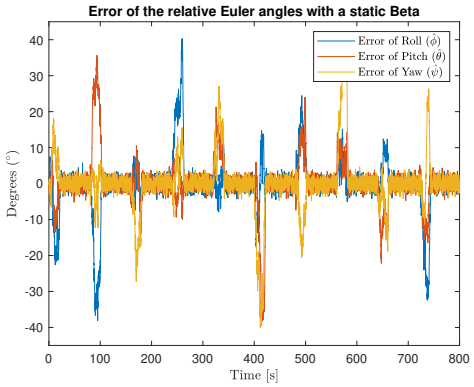


(b) Convergence of the bias estimate, where the constant where the constant gyroscopic bias is $[-0.08 \quad 0.07 \quad 0.03]$ rad/s.

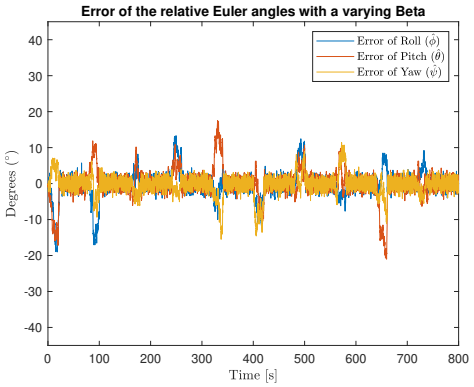
Figure 5-9: The first 3000 samples that show the convergence of the gyroscope bias estimate for the first and second sensor with a mean and three standard deviations.

.

that there is movement. The reason behind this significant decrease in the second simulation stems from equation Eq. (5-15) and Eq. (5-16). Here we see that the value of β determines how much we allow the joint kinematic constraint calculations to influence the final filtered estimate. Due to the no-movement taking place, the constraints used to create a correct estimate will be less reliable. Therefore, it would be safer to trust the estimates of the gyroscope more. The difference between the two simulations is depicted in Figure 5-10, error in the three Euler angles is shown during the last Monte Carlo simulation.



(a) No change in β occurs throughout the entire simulation, where $\beta = \sqrt{3}\sigma_\omega$. The choice of β purely depends on the amount of drift that we expect from the gyroscope measurement integration.

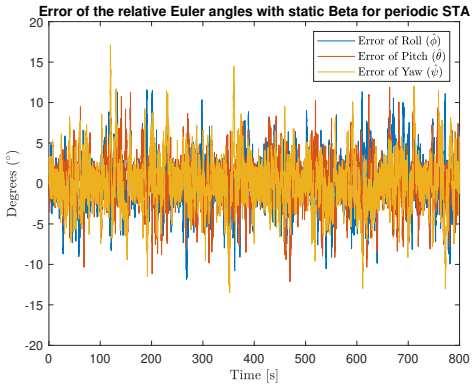


(b) A constantly changing β occurs throughout the entire simulation where $\beta = \sqrt{3}\sigma_\omega$ for when movement occurs and $\beta = 0.01\sqrt{3}\sigma_\omega$ for when no movement occurs.

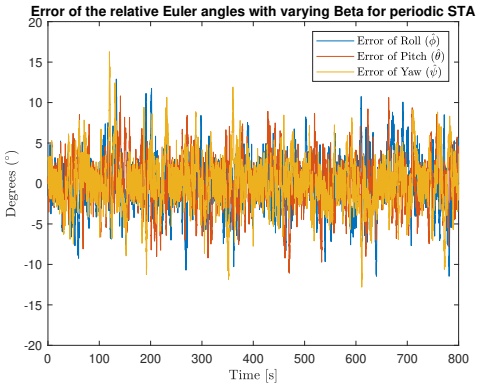
Figure 5-10: The difference in error during the last Monte Carlo simulation of simulation one and two. In both simulation we remove periodically remove all simulated movement and apply a static versus a dynamic β.

.

The results regarding the accuracy during the 100 Monte Carlo simulations of simulation one and two are given in Table 5-6. Here we observe a clear decrease in the RMSE and the standard deviation of the error for all three Euler angles when we apply a varying value for β. For the second case, we will alter β according to the periodic property of the STA, as shown in Figure 5-6. We simulated high levels of STA, as well as outliers for which the procedure was mentioned before. Again, we choose to lower the value of β, but this time it will be lowered by $\beta = 0.6\sqrt{3}\sigma_\omega$, for when the STA occur. It should be noted that we did experience a significant increase in RMSE when the value for β was set lower than 0.6.



(a) No change in β occurs throughout the entire simulation third simulation. Here, $\beta = \sqrt{3}\sigma_\omega$, which is chosen depending on the amount of drift that we expect from the gyroscope measurement integration.



(b) A constantly changing β occurs throughout the entire simulation where $\beta = \sqrt{3}\sigma_\omega$ for when no high level STA occur and $\beta = 0.6\sqrt{3}\sigma_\omega$ for when we observe high levels of STA.

Figure 5-11: The difference in error during the last Monte Carlo simulation of simulation one and two. During Both simulations high levels of STA occur, we test the algorithm for accuracy when static β is chosen versus a dynamic β.

.

In Table 5-6, the results for the third and fourth simulation are shown, where we change the value for β according to the periodical appearance of STA. We observe that the decrease in RMSE is minimal. Therefore, we can conclude that a varying β can be used when high levels of STA are expected. Nevertheless, the results due to using this method are to be expected minimal. This minimal increase in accuracy is because, during this simulation, both the gyroscope signal and accelerometer signal are corrupted. The gyroscope measurements are corrupted by noise, which was also the case for when no movement occurred, but additionally, the accelerometer measurements are severely corrupted by high levels of STA. This is why the effects of having a varying β are more fruitful during the simulations when no movements occurred since, during those simulations, the accelerometer signal is not corrupted severely.

5-4 Experimental Results via an Optical Reference System

In this research project, we did not have access to an optical reference system due to the Covid-19 virus. Luckily, we were able to obtain optical reference data along with corresponding IMU data from MSc. I. Weygers from the KU Leuven, who is the author of [61]. We have acquired three different data-sets, two of which are regular gait analyses, and one data-set

Simulation		Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	Yaw ($\hat{\psi}$)	$\sigma_{\hat{\phi}}$	$\sigma_{\hat{\theta}}$	$\sigma_{\hat{\psi}}$
1	No Movement Static β	6.7231°	6.3519°	6.5718°	1.2506°	1.0058°	1.2388°
2	No Movement Dynamic β	3.8175°	3.8390°	3.7379°	0.6239°	0.5565°	0.6505°

Simulation		Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	Yaw ($\hat{\psi}$)	$\sigma_{\hat{\phi}}$	$\sigma_{\hat{\theta}}$	$\sigma_{\hat{\psi}}$
3	High STA Static β	3.0001°	2.9958°	3.0128°	0.1193°	0.1031°	0.1211°
4	High STA Dynamic β	2.8666°	2.8527°	2.8758°	0.1323°	0.1145°	0.1290°

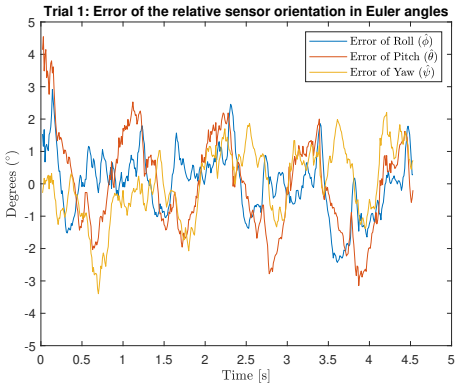
Table 5-6: The results for when a varying β parameter is applied; in the first two simulations we make use of a varying β to counter the negative effects on the accuracy due to no movement and observe a great increase in accuracy. During the third and fourth simulation a varying value for β is applied to counteract the negative effect of high levels of STA, where we observed a minimal increase in accuracy.

is of a participant performing a forward lunge movement [17]. The IMU data-sets were obtained via two MTw Awinda IMUs from Xsens. The optical motion analysis reference system consisted of 13 infrared cameras (VICON Vero, Vicon Motion Systems Ltd). Both systems were measured at a sample rate of 100 Hz. Hardware time synchronization was used to capture inertial measurements and marker trajectories simultaneously. For the experimental results in [61], eleven healthy human participants with no history of knee surgery before the testing participated in the experiments conducted for [61]. It should be noted that for the experimental results in this section, it is unknown from which of the eleven participants we obtained the data. Additionally, it should be noted that these data-sets do not have the IMUs placed on the places, as we stated in the introduction for the double sensor case (one on the wrist and one on the upper arm). However, the results obtained from these data-sets correspond to when they are placed upon the wrist and upper arm. For the first experiment, the IMUs were attached latero-cranial on the shank and on the lateral side at mid-distance on the thigh via Velcro strips [61]. For the second and third experiments, the sensors are placed more anterior, i.e., to the front. This causes more movement of the sensor, and thus higher levels of STA occur during these trials. Special 3-D printed cases were created that contain reflective marker clusters, which could house the MTw Awinda sensor. This allowed for simultaneously capturing data from the IMUs as well as data from the optical reference system. At the start of each experiment, the participants were asked not to move for five seconds. Subsequently, they were asked to perform either the forward lunge movement or a walk. During the first and second experiments, low and moderate levels of STA are observed, while in the third, high levels of STA are observed, which is due to the movement of the muscle during the forward lunge movement and the placement of the sensors. To compare an orientation estimation obtained via sensor measurement $\left(\hat{q}_t^{GS_1}, \hat{q}_t^{GS_2}\right)$ to the orientation obtained via the optical reference system $\left(q_t^{OM_1}, q_t^{OM_2}\right)$, the constant misalignment $q^{M_2S_2}$, and $q^{M_1S_1}$ are estimated from the relative orientation references $q_t^{M_1M_2}$, and relative orientation estimates $\hat{q}_t^{S_1S_2}$. The theory behind this was already explained in Section 2-3-3, with the

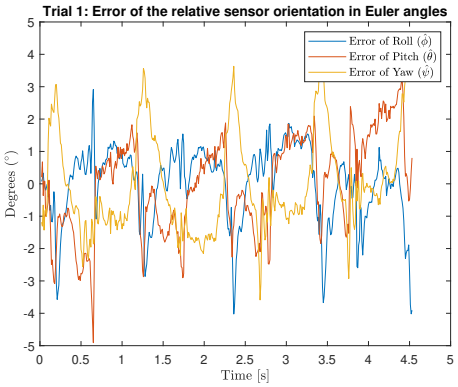
objective function, as

$$\begin{aligned} q^{S_1M_1} \odot q_t^{M_1M_2} \odot q^{M_2S_2} &\approx q_t^{S_1G} \odot q^{GO} \odot q^{OG} \odot q_t^{GS_2}, \\ q_t^{M_1M_2} \odot q^{M_2S_2} &\approx q^{M_1S_1} \odot q_t^{S_1S_2}. \end{aligned} \tag{5-31}$$

Together with the data-sets that were granted, we also obtained the results per data-set when the proposed MEKF method from [61] was used. This allows us to compare our proposed method versus the MEKF approach. In [61], the first five seconds are used to empirically define the noise variance to create the covariance matrices for the MEKF. The initial orientations were set to $[1 \quad 0 \quad 0 \quad 0]^\top$ and the initial process covariance matrix was set as a six by six identity matrix. The measurement noise covariance matrix and the covariance matrix for the error induced by the joint acceleration rotation were both set to be a three by three identity matrix. The values for r_1 and r_2 were estimated via the method presented in [52], which minimized the sum of squared errors. For all three trials, we estimate the length from the jc to the IMU with the method that minimizes the sum of absolutes as proposed in [40] and numerically tested in Section 5-3-2. We choose this method since we have already proven that this would be the most suitable method for experimental trials, where STA and outliers may occur. Simultaneously, we have approximately used the first five seconds of each trial to obtain the gyroscope's covariance, from which we determine the values for β. We found $\beta = 1.72 \cdot 10^{-1}$, $\beta = 1.65 \cdot 10^{-2}$ and $\beta = 1.89 \cdot 10^{-2}$ for the first, second and third trial respectively.During the first experiment, our proposed complementary filter performs better than the proposed MEKF from [61]. However, it should be noted that when no STA would occur, it is to be expected that the MEKF from [61] would outperform our proposed complementary filter. Since MEKF are known to be more accurate in comparison to complementary filters. The RMSE values in three directions are shown in Table 5-8. Both errors of the relative sensor orientation in Euler angels are shown in Figure 5-12. Here it is observed how the error develops over time, and we see, although the errors deviate slightly from each other, they have the same pattern.



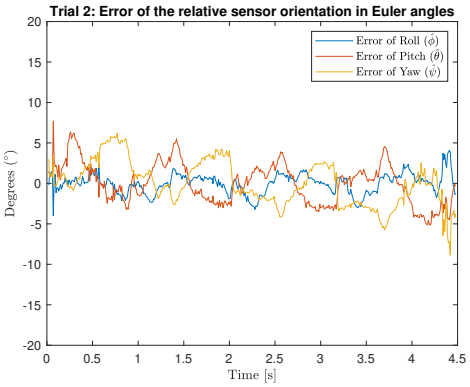
(a) The error in the relative sensor orientation for our proposed complementary filter of Algorithm 4 when low levels of STA occur during the first trial.



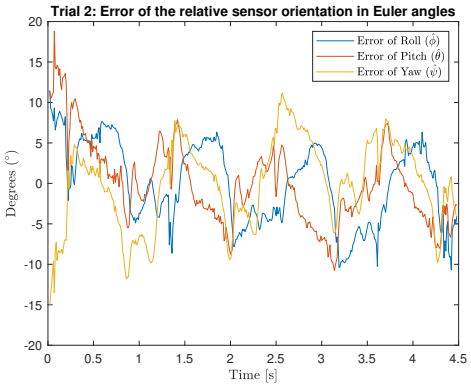
(b) The error in the relative sensor orientation of the MEKF method presented in [61] when low levels of STA occur during the first trial.

Figure 5-12: Our proposed complementary filter presented in Algorithm 4 versus the MEKF from [61]. We observe that our proposed method performs better than the proposed MEKF from [61] when low level of STA occur.

.



(a) The error in the relative sensor orientation for our proposed complementary filter of Algorithm 4 when moderate levels of STA occur during the second trial. Here we clearly see that the algorithm is robust against this disturbance.



(b) The error in the relative sensor orientation of the MEKF method presented in [61] when low levels of STA occur during the second trial. Here we observe that the algorithm's performance start to degrade due to the increasing levels of STA.

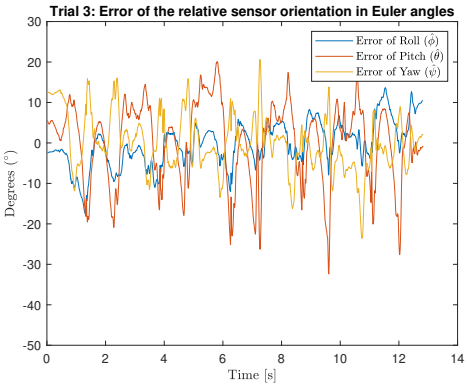
Figure 5-13: The second trial where moderate levels of STA occur. We observe a clear difference in performance when comparing our proposed complementary filter versus the MEKF from [61], our proposed method performs better.

.

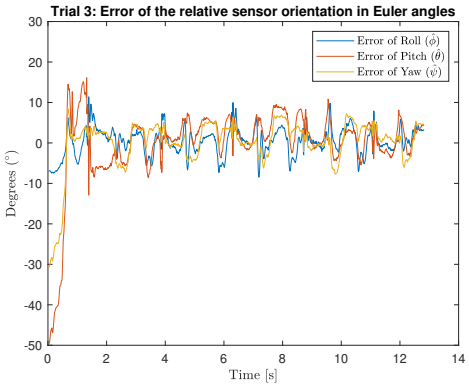
During the second trial, the sensors were placed more anterior before performing the gait analyses, which causes moderate levels of STA to occur during this trial. We see in Table 5-8 that our proposed algorithm can reject the disturbance in the form of STA better than the proposed MEKF from [61], and thus performs better. This is also visually represented in Figure 5-13, where the error in the relative sensor orientation is shown.

In the third and last trial, which contains movement-data from a series of forward lunge motions, high levels of STA are present. The high levels of STA are reflected in the results, as shown in Table 5-8. We observe that both our proposed complementary filter and the MEKF from [61] cannot create a reliable relative sensor orientation estimate. The decrease in performance was expected if we take into account the numerical simulations performed in Section 5-3-4, here it was already shown that when high levels of STA would occur, the estimation of the relative sensor orientation would become unreliable. In Figure 5-14, the error is depicted for both algorithms, where we observe that neither our proposed complementary filter can generate an accurate measurement as well as the proposed MEKF from [61].

What might be of a positive influence on our purposed method's accuracy is the elimination of the gyroscope bias, as was presented in Section 2-2-5 and numerically tested in Section 5-3-4. We compare the three trials where during the first three, no bias estimation is used, and during the last three, the bias estimation method from [31] is applied. The values for β during the three trials are unchanged. The results are shown in Table 5-7. For the first trial, we observe that the bias estimation has a small effect on the accuracy of the estimation algorithm. We observe that the estimation of the bias leads to a small decrease regarding the accuracy, which may indicate that the bias was of little influence. During the second trial, we observed a clear enhancement regarding performance, which led us to believe that the bias was of significant influence during the second trial. In the third and final trial, we observed a considerable decrease regarding performance. The explanations for this would be that the bias



(a) The error in the relative sensor orientation for our proposed complementary filter of Algorithm 4 when high levels of STA occur during the third trial. It can be observed that our proposed method experiences difficulties with the retention of a correct relative sensor orientation.



(b) The error in the relative sensor orientation of the MEKF method presented in [61] when high levels of STA occur during the third trial. Here we observe that the algorithm's performance, similar to ours has difficulties with the creation of a correct estimate of the relative sensor orientation.

Figure 5-14: The third trial, where high levels of STA occur during the performance of a forward lunge movement in combination with the placement of the sensors, which are placed more to the front. Both estimation algorithms experience trouble regarding robustness against these high levels of STA.

.

estimation algorithm is not able to correctly estimate the bias due to the high levels of STA; therefore, a wrongly estimated bias is being subtracted from the gyroscope measurements in the third trial, leading to inaccurate measurement, which corrupts the estimation of the relative sensor orientation. From this, we can conclude that the bias estimation algorithm from [31], presented in Section 2-2-5, can be used when low or moderate levels of STA are to be expected.

Lastly, we can search for improvement regarding the accuracy of the third trial; this might be possible using a varying β. When the forward lunge movement is performed, the moments are easily detectable due to the movement generating spikes in the accelerometer measurements. Hence we can detect when a forward lunge is performed, and with this information, we can detect when high levels of STA occur. During the moment when no forward lunge movement is being performed, and thus when the levels of STA are low or non-existent, we choose a $\beta = 1.89 \cdot 10^{-2}$. Subsequently, when a forward lunge movement is preformed we choose to lower the value to $\beta = 0.6 \cdot 1.89 \cdot 10^{-2} = 1.13 \cdot 10^{-2}$. From this we obtained 5.4943°, 9.9747° and 6.8765° as values for the RMSE in the roll, pitch and yaw respectively. This means that the use of varying value for β has led to a slight decrease in RMSE when compared to the values obtained without varying β, as can be seen in Table 5-8. From this, we can conclude that the use of varying value for β can lead to an increase in accuracy and should be studied in the future. As mentioned in the numerical simulations, the effects are minimal due to the severely corrupted accelerometer measurements. From the experimental results that are obtained in this section, we can state that our proposed complementary filter can be used as a suitable extension for prosthesis research since it is proven to be robust against most disturbances that occur during a real-world setup. Additionally, we know that the complementary filter is

Trial			Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	Yaw ($\hat{\psi}$)
1	Low STA	without bias estimation	0.9722°	1.4793°	1.0895°
		with bias estimation	0.9863°	1.5734°	1.0777°
2	Moderate STA	without bias estimation	1.2337°	2.5043°	2.7399°
		with bias estimation	1.1123°	1.9120°	2.2068°
3	High STA	without bias estimation	5.5601°	9.9807°	6.9118°
		with bias estimation	30.8406°	28.8579°	56.8299°

Table 5-7: The results for the three trials, when the bias estimation method from [31] is used. It shows that the bias estimation method is helpful when low or moderate STA levels are expected. During the forward lunge movement, which induced high levels of STA upon the accelerometer measurements, the bias estimate became severely corrupted, which caused a significant decrease in performance.

computationally efficient, making it an attractive research tool for prosthesis research.

Trial			Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	Yaw ($\hat{\psi}$)
1	Low STA	Algorithm 4	0.9722°	1.4793°	1.0895°
		MEKF from [61]	1.5582°	1.4667°	1.1852°
2	Moderate STA	Algorithm 4	1.2337°	2.5043°	2.7399°
		MEKF from [61]	5.3134°	5.0609°	4.9964°
3	High STA	Algorithm 4	5.5601°	9.9807°	6.9118°
		MEKF from [61]	6.2792°	9.6570°	3.3983°

Table 5-8: The comparison between the results that are obtained when our proposed method is used versus the proposed MEKF approach from [61]. The values indicate that our proposed method remains robust during the first two trials, while the method from [61] performs less well when disturbances in the form of STA are introduced. When high levels of STA are introduced, both algorithms have trouble regarding performance.

Chapter 6

Discussion

In the introduction, we mentioned three options regarding the sensors to improve artificial upper limb analysis further. In this section, we will discuss all significant findings per option that have been created during this project and discuss the meaning behind those findings consecutively. After this, we will cover how our findings relate to the work of others, where we will discuss similarities and differences. At last, we will discuss the limitations of our findings.

- Extract more information within state-of-the-art methods that require an accelerometer.

The first extensions to enhance prostheses evaluation and research were created in Chapter 3, here we showed how to create a tilt estimation algorithm with only the use of an accelerometer. During the numerical simulations and real-world experiments, we found that when the sensor is only rotated, and no external linear accelerations are exerted upon the accelerometer, the tilt estimation algorithm works well. We also found that when external linear accelerations are exerted upon the accelerometer, the assumed measurement model is violated, and the tilt estimation algorithm becomes unreliable. This entails that accelerometer-only tilt estimation is not suitable for prosthesis research.

Another extension that we created stems from the desire to bypass the use of activity counts (AC) from ActiGraph, which are used in [16]. We developed a new method to create a vector magnitude (VM) by looking at the length of the gravity and accelerometer measurement vector. We also introduced a novel scoring system that allows us to observe what type of intensity is used while a movement is performed. From the VM and scoring system, we were able to create histograms to indicate the level of contribution per arm and spiral plots, where we chose to plot two spiral plots in one figure. Because of this, we created a new way to observe what type of movement is conducted by the arms, while at the same time enabling us to observe the intensity of the movement.

A limitation of the new method for creating VM is the choice regarding the standard deviation bounds that we use. We saw that when low bounds are used, we can obtain accurate

classifications for bilateral use, but practical difficulties regarding the classification when both arms are at rest occur. When a higher bound was chosen, we observed accurate classification for the situation where both arms are at rest, but experienced difficulties for the classification of bilateral and unilateral movement.

- Add different sensors in the form of a gyroscope, which will translate to using an Inertial Measurement Unit (IMU) instead of only using an accelerometer.

Extending the existing approach with a gyroscope opened up possibilities to use Sensor Fusion (SF) algorithms to estimate the inclination accurately. Based on the proposed algorithm from [24], we created an extension for prosthesis evaluation and research, which allowed for inclination estimation that remained accurate despite assumed measurement model violations and accelerometer outliers. We showed that our findings corresponded with the ones obtained in [24] and determined that inclination estimation via Algorithm 3 is deemed as a suitable extension for prosthesis research.

We also created a new method for the creation of VM from gyroscope measurements and a corresponding scoring system that can indicate the intensity of movement. This gives prosthesis researchers an additional option to choose from and therefore removes a particular part of the limitations regarding prosthesis research that currently exists. Similar to the accelerometer method, we found the same limitations and problems when we created the VM and activity scores with the gyroscope measurements. By increasing the bound's value, we can classify the moments when both arms are at rest more accurately but lose the capability to classify bilateral movement accurately. This results in an increase in the wrong classification of unilateral movement and both arms at rest. By decreasing the value of the bounds, the opposite occurs; when the value of the bounds is decreased, we increase the accuracy in classifying bilateral movement correctly but lose accuracy in classifying both arms at rest correctly. This causes a trade-off between the choice of the bound's value for the classification of movement.

- Adding multiple sensors, whereby several Inertial Measurement Units (IMUs) are used to extract information for upper limb analyses.

Our last extension uses two IMUs placed upon two adjacent segments connected via a spherical joint. Our algorithm integrates gyroscope data and corrects the resulting joint kinematics for integration drift using shared information in the accelerometer and gyroscope measurements of sensors placed on connecting segments, using a complementary filter. The complementary filter can efficiently process large quantities of data, opening up for online use and use in a non-clinical environment. Before testing our proposed complementary filter, we thoroughly tested two methods to estimate the distance from the joint center (jc) to the IMU. The first method is from [52], which uses a Gauss-Newton algorithm to minimize a sum of squared errors cost function. The second method is from [40], which minimizes the sum of absolute errors. The optimization method that we chose, to minimize the sum of absolute errors objective function, was a backtracking line search in gradient descent in combination with stopping criteria, such that the algorithm would stop iterating if the decrease in the cost function would be smaller than 0.1%. We tested both methods and found that when the

approximation of the angular acceleration is not low-pass filtered, the estimation results will become inaccurate, similar to [52] we tested the estimation algorithms with an additional low-pass filter on the approximation of the angular acceleration. After this, we observed that the results for both methods became better, but were still inaccurate. This had to do with the high levels of simulated gyroscopic noise. We discovered that with a low-pass filter on the gyroscope measurement signal, we were able to overcome the problems regarding the accuracy and hence obtained an accurate estimation of the distance vector. When additional accelerometer outliers were added, the distance vector's estimates became unreliable when the first method from [52] was used, while the method from [40] showed promising robustness properties. We also tested the second method against low, moderate, and high levels of Soft-Tissue-Artifacts (STA), which is a common disturbance that can occur during prosthesis research. After testing, we found that the method remained reliable until high levels of STA were introduced. Hence, we found that this method should be used for prostheses evaluation and research when relative sensor orientations want to be estimated.

After testing the two methods to obtain the distance vector, we tested our proposed filter numerically and experimentally. Our simulations showed that the algorithm remains robust against common real-world disturbances, among which are accelerometer outliers, and different levels of STA. We observed a clear degradation in performance when high levels of STA were induced upon the accelerometer. We also introduced a gyroscope bias estimation algorithm from [31], which we extended for a double sensor system. Again, we observed that when high levels of STA occurred, the bias estimation algorithm became unreliable. After this, the use of a varying value for β was tested. Two situations were created, where the first contains moments when no movement occurs, which is a violation of the assumed measurement model, and the second is when high levels of STA periodically occur. The value for β was lowered whenever no movement occurred, and we saw an increase in accuracy compared to when a fixed value for β was used. For the second occasion, where we lowered the value of β when high levels of STA occurred, a small increase in accuracy was observed, the increase is small due to both signals from the IMU being corrupted. The gyroscope measurements are corrupted by high noise levels, while the accelerometer measurements are corrupted severely by the high levels of STA.

During the experimental results, we were able to test our proposed method versus the Multiplicative Extended Kalman Filter (MEKF) method that is proposed in [61]. We tested both methods during three trials, where each trial contained an increasing amount of STA levels. We observed that during the trials where low and moderate levels of STA occur, our complimentary filter remained robust and performed more accurately than the proposed MEKF of [61]. When high levels of STA occurred, both estimation methods could not create an accurate relative sensor orientation estimate. Our proposed method is deemed as a suitable extension for prosthesis research to estimate the relative sensor orientation.

Chapter 7

Conclusion

In this work, we present several extensions that can be used to enhance prostheses' evaluation and research. The starting point was the analysis of the high rejection rates regarding prostheses' use, which led to the increased importance of assessing the amputees' use of the prosthesis. We found that there was potential for improving and further enhancing the development of prosthesis evaluation and research methods. Hence, we formulated three scenario's that could serve as possible directions for prostheses research enhancement. We created a single sensor case and a double sensor case. The single sensor case, which could be further distinguished into two parts: one where only an accelerometer is used and the case where an additional gyroscope is used. The double sensor case was defined as having two Inertial Measurement Units (IMUs) per arm, where one is placed upon the wrist and the other upon the upper arm.

Extensions for the first case, which only concerned the use of an accelerometer, were shown in Chapter 3. Here we showed how Sensor Fusion (SF) might aid in the analysis of the upper limb, but we also showed and discussed limitations regarding the created extension. The extensions were split up into two parts; the extension to create an estimate for the sensor's inclination in the form of a unit-quaternion, and an extension to bypass the use of activity counts (AC) from ActiGraph. We can answer the following research question that was posed at the beginning of the introductory chapter due to our findings.

- ***How can we extract more information within state-of-the-art methods?***

  We attempted to extract more information with an accelerometer-only approach in two ways. The first was the tilt estimation algorithm presented in Algorithm 2 in Chapter 3, where we showed that we could create a correct tilt estimate only when the sensor is rotated, and no external accelerations are induced upon the accelerometer. External accelerations create violations in the assumed measurement model, which leads to considerable errors. Hence, tilt estimation with an accelerometer-only scenario is not a possible extension for prostheses research and evaluation.

  The second attempt to extract more information from state-of-the-art methods draws inspiration from [29] to create a novel method, from which we can create a vector

magnitude (VM) from the accelerometer measurements. Simultaneously we created a new scoring system, which can be used as an extension within state-of-the-art methods. It should be noted that there is a limitation regarding this method, which consists of the standard deviation bounds that can cause the wrong classification of a certain type of movement.

In Chapter 4, we treat the extensions for the second case, where an additional gyroscope can be used. An additional gyroscope opens up possibilities to use more advanced SF techniques to acquire the sensor's inclination. Additionally, we observed that the tools from [16] and the tools created in Section 3-4 are limited since they only treat linear movement. With those methods, the acceleration is analyzed while ignoring all rotational movements. In Chapter 4, the extension for the Activity Scoring System is created where we include information of the rotational velocity from the arms. From the information obtained in Chapter 4, we can answer the following question.

- ***How can we enhance prostheses evaluation when we add different sensors?*** When different sensors were added in the form of an additional gyroscope, we were able to create a magnetometer free tilt estimation using the proposed algorithm from [24]. This algorithm, which is shown in Algorithm 3 remains robust during violations of the assumed measurement model, high levels of gyroscope noise, and accelerometer outliers. Therefore, Algorithm 3 is deemed suitable as a tool to enhance existing prosthesis research and evaluation.

  Additionally, we were able to create a new form of VM with a corresponding scoring system, which considers the rotational velocity and, thus, rotational movement, instead of linear movement. The limitations of this method consist of the bounds that are created from the standard deviations. When the bounds are low, the bilateral movement classification is done accurately, but the classification of both arms at rest is done consistently incorrect. When the bounds are increased in value, the bilateral movement classification is done inaccurate, while the classification for both arms at rests is done accurately.

In Chapter 5, we present a novel algorithm for the estimation of the relative sensor orientation. IMUs can be attached to body segments, and their absolute orientations can be estimated. However, the heading part of such orientation estimates is known to drift over time, resulting in drifting joint kinematics. Chapter 5, described a novel joint kinematic estimation method that incorporates the connection between adjacent segments within a SF algorithm that remains accurate in the vicinity of common real-world disturbances. Our algorithm integrates gyroscope data and corrects the resulting joint kinematics for integration drift using shared information in the accelerometer and gyroscope measurements of sensors placed on connecting segments, using a complementary filter. The complementary filter can efficiently process large quantities of data, opening up possibilities for real-time usage and use in a non-clinical environment. Due to our findings in Chapter 5, we can answer the following research question.

- ***How can we enhance prostheses evaluation when we add more different type of sensors?***

Prosthesis evaluation can be enhanced by creating a novel method to estimate the relative sensor orientation, which is robust and computationally efficient. This is done by using the constraints imposed by the human body itself in the form of rigidly connected segments, accompanied by a complementary filter. We were able to create an accurate estimation for the relative sensor orientation between the two sensors. Therefore, it could be concluded that this method would work if the sensors were placed on the wrist and upper arm. We have proven that this method is robust against high levels of gyroscopic noise, accelerometer outliers, and up to moderate levels of Soft-Tissue-Artifacts (STA). What we did find is that when too much STA occurs, the estimation algorithm is not able to perform a correct estimation anymore and thus becomes unreliable. We tested our proposed method versus the Multiplicative Extended Kalman Filter (MEKF) method proposed in [61] and showed that our estimation algorithm performs better in the vicinity of low and moderate levels of STA. Therefore, our proposed approach is deemed suitable as another extension to existing prosthesis evaluation and research methods.

To answer our first research question, which served as the fundamental question during this whole thesis project, we were required to develop several different methods, techniques, and strategies to extract new clinically relevant information that can be used in prosthesis evaluation and research. Because we answered the three previous questions accordingly, we can answer our main question for this research.

- ***What is the best way to enhance the analyses of upper limb prostheses?*** In this research, we found that the best way to enhance upper limb prostheses' analyses is to start incorporating gyroscopes into the research process. This can either be in the form of the single sensor case, where the participant wears an Inertial Measurement Unit (IMU) around each wrist, or the double sensor system, where the participant is required to wear two IMUs per arm, one on each wrist and one on each upper arm. The additional gyroscope will allow us to use one of the many SF techniques, as discussed in this report. All methods are created such that large quantities of data can be processed in a computationally efficient way.

A recommendation for future studies could be to look at the current state of affairs regarding the classification for when a VM is classified as zero. As seen in Chapter 3 and 4, a bound that consists of a number of standard deviations is currently used, but we experienced that this will lead to a trade-off between accuracy in the classification of bilateral movement and movements when both arms are at rest. Another recommendation for future studies is to research the effects of high levels of STA, and even more important, how to create some form of robustness against it. A third recommendation would be to create a specif estimation algorithm for the Self Grasping Hand (SGH), because of the clicking noise, which the SGH creates when it is closed, it might be possible to detect how many clicks are performed. We know that a certain amount of clicks correspond to a fixed position that the hand is in. With this new information, we might be able to estimate the hand's relative sensor orientation with respect to the IMU that is placed upon the wrist without using a sensor on the hand itself. We would also like to recommend future studies to explore the benefits of a varying value for β; in this research, we only showed two occasions for which we showed the results

of having a dynamic value for β. During the occasion when high levels of STA periodically occurred, we only observed a minimal increase regarding accuracy. Therefore, this should be explored further. Our final recommendation for future studies would be to create more experimental data-sets to test Algorithm 4 more extensively. Due to the Covid-19 virus, we could not create data-sets of ourselves, and therefore only used three experimental data sets, which were granted by MSc. I. Weygers. In Appendix A, a draft version of a paper is shown, which discusses the main concepts behind Chapter 5. For our final paper, we will most likely create more experimental data.

Appendix A

Draft Paper

The work presented in this thesis has lead to a first draft of a paper, this paper presents the concepts and ideas of the proposed complementary filter from Chapter 5.

Fast relative sensor orientation estimation in the vicinity of real-world disturbances

Evan Remmerswaal[1], Ive Weygers[2], Gerwin Smit[3] and Manon Kok[1],

***Abstract*— We present a novel algorithm for the estimation of the relative sensor orientation. Inertial measurement units can be attached to body segments, and their absolute orientations can be estimated. However, the heading part of such orientation estimates is known to drift over time, resulting in drifting joint kinematics. This study proposes a novel joint kinematic estimation method that incorporates the connection between adjacent segments within a sensor fusion algorithm that remains accurate in the vicinity of common real-world disturbances. Our algorithm integrates gyroscope data and corrects the resulting joint kinematics for integration drift using shared information in the accelerometer and gyroscope measurements of sensors placed on connecting segments, using a complementary filter. The complementary filter can efficiently process large quantities of data, opening up possibilities for real-time usage and use in a non-clinical environment.**

I. INTRODUCTION

The capability to know joint kinematics in outside laboratory environments is clinically relevant [1]. Body-worn motion sensors are now used more often as a non-invasive, objective, and accurate way to observe the movement of a subject [2], [3]. These data-sets are mostly recorded over a more extended period, e.g., one or two weeks [4]. Therefore, capturing joint kinematics in a computationally efficient way is sought after.

For the most basic orientation estimation, inertial sensor fusion algorithms rely on the knowledge obtained from the gyroscope, the accelerometer, and the magnetometer. In a basic orientation estimation setup, the sensors only rotate and will not translate. The accelerometers will then only measure the local gravity vector since no linear acceleration is exerted on the sensor [5]. From this measured gravity vector, it is possible to deduce information about the inclination of the sensor. Simultaneously, the magnetometer complements the other inertial sensor measurements. This is because it provides information about the heading of the sensor. In other words, it provides information about the orientation around the gravity vector, and this information cannot be deduced from accelerometer measurements only. The magnetometer can provide heading information all around the world except at the magnetic poles since the magnetic field is vertical [5].

[1]Evan Remmerswaal and Manon Kok are with the Delft Center for Systems and Control, Delft University of Technology, the Netherlands. Email: j.w.g.remmerswaal@student.tudelft.nl, M.Kok-1@tudelft.nl

[2]I. Weygers is with the KU Leuven at the department of Movement and Rehabilitation sciences, KU Leuven, Belgium. Email: ive.weygers@kuleuven.be

[3] Gerwin Smit is with the TU Delft at the department of Medical Instruments Bio-Inspired Technology, Delft University of Technology, The Netherlands. Email: g.smit@tudelft.nl

The relative sensor orientation between two sensors on adjacent segments, as seen in Fig. 1, could be obtained from two correct absolute orientation estimates. The problem with this method is that magnetic disturbances often occur in an outside laboratory environment, which causes the magnetic field to be mismeasured. The estimation of the relative sensor orientation, while omitting the magnetometer has been studied before. In [1], a novel joint kinematic estimation method is proposed, which tightly incorporates the connection between adjacent segments within two different approaches. Both an optimization-based smoothing and a filtering approach were implemented to obtain drift-free joint kinematics. The authors of [6] propose another optimization-based solution that does not require a magnetometer to be used. The method allows for the natural inclusion of biomechanical constraints, handling nonlinearities, and using all data to obtain the relative sensor orientation estimate.

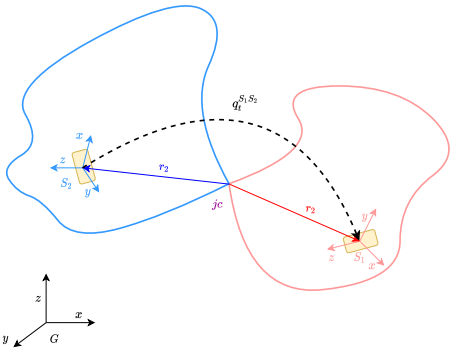


Fig. 1: Relative sensor orientation between two Inertial Measurement Units that are placed upon adjacent segments.

Although current methods allow for accurate relative sensor orientation estimation, it is known that the optimization-based methods inherently deal with high computational complexity. Simultaneously, robustness against disturbances that commonly occur during joint kinematic estimation can cause the estimation algorithm's accuracy to decrease rapidly. This was already shown in [7], where the effect of Soft-Tissue-Artifacts (STA) was explored. The STA is a common disturbance that arises because human body segments are non-rigid. The sensors are placed on soft tissue, which can lead to the violation of the assumption that the sensor position with respect to the joint is fixed at all times. These types of disturbances are known as Soft Tissue-Artifacts (STA) [8]. Another form of outliers that should be taken into account is accelerometer outliers. In the real world, these can

be caused by the sensor or the body that the sensor is placed upon banging or ticking against a hard surface; this will then cause a spike in the accelerometer measurement.

In this paper, we present a novel algorithm, in the form of a complementary filter, that can compute joint kinematics without using a magnetometer, which is computationally efficient. This type of algorithm is known to be robust against outliers, and we also study and explore its robustness against STA [9].

II. Sensor modeling

We are interested in estimating the relative sensor orientation of two sensors, which are placed upon adjacent segments. To this end, we will estimate the orientation of each sensor. We will use the angular velocity to provide information about the change in orientation and use joint kinematic constraints to remove drift in the relative sensor orientation. These joint kinematic constraints can come from two consecutive segments of a human body that can be modeled as two adjacent rigid segments, which are connected via a spherical joint [1]. Spherical joints allow for no restriction regarding rotational Degrees of Freedom (DoF) and thus allow for rotational movement in all directions. Hence in our model, each segment consists of an Inertial Measurement Unit (IMU) with a corresponding sensor coordinate frame, which we denote as S_i, with index i (where $i = 1, 2$). The sensor coordinate frame has its origin lay at the center of the accelerometer triad, which is aligned with its casing. We model the measurements of the gyroscope at each time instance t as

$$y_{\omega,t}^{S_i} = \omega_t^{S_i} + e_{\omega,t}^{S_i}, \tag{1}$$

where $\omega_t^{S_i}$ denotes the angular velocity of the sensor and $e_{\omega,t}^{S_i}$ the measurement noise of the i-th. Here $e_{\omega,t}^{S_i}$ is a zero-mean Gaussian noise, distributed as $\mathcal{N}(0, \Sigma_\omega)$. It should be noted that (1) can be optionally expanded with a gyroscope bias.

For the accelerometer measurement model, we assume that the accelerometer measures a gravity g^G component, accompanied by an additional linear acceleration of a^G. Here the superscript G denotes the local geographic frame in which we want to navigate. The accelerometer measurements $y_{a,t}^{S_i}$ are hence modeled as

$$y_{a,t}^{S_i} = R\left(q_t^{S_iG}\right)\left(a^G - g^G\right) + e_{a,t}^{S_i}. \tag{2}$$

We choose to represent the orientation at time t by a unit-quaternion denoted as $q_t^{GS_i}$. Here the superscript GS_i denotes that the quaternion represents a rotation from the i-th sensor frame to the global frame. The operation $R\left(q_t^{GS_i}\right)$ converts the unit-quaternion to a rotation matrix, describing a similar rotation. Rotation matrices are square matrices, with real entries. More specifically they can be characterized as an orthogonal matrix with determinant equal to one. Hence, the inverse operation in (2) is given by $R\left(q_t^{GS_i}\right) = \left(R\left(q_t^{S_iG}\right)\right)^\top$. Furthermore, we assume that $y_{a,t}^{S_i}$ in (2) has unit norm and $g^G = [0 \quad 0 \quad 1]^\top$.

III. Joint Kinematic Modeling

We know that a common point that serves as the intersection between two rigid body segments, named the joint center (jc), should have a common expression for its acceleration $a_{\text{jc},t}^G$. Due to the measurement noises and model inaccuracies, errors arise in the rotation from the first and second sensor frame to the global frame. Hence, the difference in the acceleration of the jc will have an induced error as

$$R_t^{GS_1} a_{\text{jc},t}^{S_1} = R_t^{GS_2} a_{\text{jc},t}^{S_2} + e_{\text{link},t}. \tag{3}$$

Here, $e_{\text{link},t} \sim \mathcal{N}(0, \Sigma_{\text{link}})$ and the jc acceleration in both the first and second sensor frame are calculated by utilizing the accelerometer measurement $y_{a,t}^{S_i}$ at a distance from the IMU to the jc by using $\mathcal{C}_t^{S_i}$ as

$$\begin{aligned} a_{\text{jc},t}^{S_i} &= y_{a,t}^{S_i} - \mathcal{C}_t^{S_i} r_i^{S_i}, \\ \mathcal{C}_t^{S_i} &= \left[y_{\omega,t}^{S_i}\times\right]^2 + \left[\dot{y}_{\omega,t}^{S_i}\times\right]. \end{aligned} \tag{4}$$

The $[\,\cdot\,\times]$ represent a cross-product matrix and $\dot{y}_{\omega,t}^{S_i}$ denotes the angular acceleration of the i-th sensor at time t. The vectors r_1 and r_2 display the distance from the IMU to the jc.

IV. Estimating the relative sensor orientation with a Complementary Filter

We update both orientations $q_t^{GS_1}$ and $q_t^{GS_2}$ together, which allows us to compensate for drift in the relative sensor orientation $q_t^{S_1S_2}$. This can be done by utilizing common information present in the gyroscope and accelerometer measurements of two adjacent IMU.

To estimate the orientation, only the integration of the gyroscope measurement is required since the angular velocity is the derivative of the orientation; hence direct integration would lead to an orientation estimate, which we denote as $q_\omega^{GS_i}$. This would indeed be correct if the initial orientation would be known, and if a perfect measurement model would exist. However, this is never the case because the gyroscope measurements contain bias and noise. The bias and noise create a less accurate measurement; they also inhibit anyone from obtaining the correct absolute orientation by just plainly integrating the measured angular velocity. If one would directly integrate the gyroscope measurement, drift will occur due to integrating the noise and bias [5]. This entails that orientation estimates and relative orientation estimate $q_{\omega,t}^{S_1S_2}$, from the gyroscope, are accurate on a short time scale but will drift over a longer time horizon. To overcome drift in the relative orientation, we can create an orientation estimate that contains information about the joint kinematics, which we denote as $q_{jc}^{GS_i}$. This relative orientation estimate, denoted by $q_{jc,t}^{S_1S_2}$, will, unlike $q_{\omega,t}^{S_1S_2}$, not drift and is accurate on a long time scale, although it is more noisy. The accuracy property of the gyroscope estimate on a short time scale and

the non-drifting property of the joint kinematics estimate can be combined by using a complementary filter in which $q_{jc}^{GS_i}$ is low-pass filtered and $q_{\omega}^{GS_i}$ is high-pass filtered as

$$\hat{Q}^{GS_i}(s) = G(s)Q_{jc}^{GS_i}(s) + (1 - G(s))Q_{\omega}^{GS_i}(s), \tag{5}$$

where s denotes the Laplace variable. The transfer function $G(s)$, denotes a first-order low-pass filter as $G(s) = \frac{1}{as+1}$, where a is the filter time-constant and $Q^{GS_i}(s)$ represents the orientation q^{GS_i} in the Laplace domain. The resulting filtered orientation $\hat{q}^{GS_i}$, is obtained via discretizing (5) using backward Euler [9] and is given as

$$\begin{aligned} \hat{q}_t^{GS_i} &= (1 - \gamma_t)\, q_{jc,t}^{GS_i} + \gamma_t\, q_{\omega,t}^{GS_i}, \\ &= (1 - \gamma_t)\, q_{jc,t}^{GS_i} + \gamma_t \left(\hat{q}_{t-1}^{GS_i} + T\omega_{q,t}^{S_i} \right), \end{aligned} \tag{6}$$

where T denotes the sampling time, $y_t = \frac{a}{a+T}$ and $\omega_{q,t}^{S_i}$ represents the angular velocity expressed in terms of a quaternion of the i-th sensor. The use of the complementary filter in this fashion was already done in [9], the novelty in this work lies in the use of two sensors in combination with joint kinematic constraints. Observing (6) it should be noted that $\hat{q}_t^{GS_i}$ does not represent a valid rotation due to the quaternion is no longer of unit length. Later on, it will become clear that the deviation in length of this quaternion is small due to the high-frequency rates of the sensors.

A. Orientation from gyroscope measurements

Similar to [9], we require the orientation from gyroscope measurements. This is done by utilizing an Euler discretization and assuming that there is a constant angular velocity between two time-samples, the dynamics of the orientation can be expressed in terms of the sampling time and the angular velocity as [5]

$$\begin{aligned} q_t^{GS_i} &= q_{t-1}^{GS_i} \odot \exp_{\mathrm{q}} \left(\frac{T}{2} y_{\omega,t}^{S_i} \right), \\ &\approx q_{t-1}^{GS_i} + \frac{T}{2} S\left(q_{t-1}^{GS_i} \right) y_{\omega,t}^{S_i}, \end{aligned} \tag{7}$$

where $\odot$ denotes the quaternion product and $\exp_{\mathrm{q}}$ denotes the quaternion version of the vector defined as

$$\begin{aligned} \exp_{\mathrm{q}}(y) &= \begin{bmatrix} \cos\alpha & v^\top \sin\alpha \end{bmatrix}^\top, \\ \alpha &= \|y\|_2, \quad v = \frac{y}{\alpha} \end{aligned} \tag{8}$$

Furthermore, for $q = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^\top = \begin{bmatrix} q_0 & q_v^\top \end{bmatrix}^\top$,

$$S(q) = \begin{pmatrix} -q_v \\ q_0 \mathcal{I}_3 - [q_v \times] \end{pmatrix} \tag{9}$$

where $\mathcal{I}_3$ denotes a 3×3 identity matrix. By observing (6) and (7), we have that the angular velocity is expressed in terms of a quaternion as

$$\omega_{q,t}^{S_i} = \frac{1}{2} S\left(\hat{q}_{t-1}^{GS_i} \right) y_{\omega,t}^{S_i} \tag{10}$$

B. Orientation from joint kinematics

With the help of (3), the estimation of the orientation from the joint kinematics can be obtained by minimizing the following objective function, with $\eta_t = \begin{bmatrix} \eta_{S_1,t}^\top & \eta_{S_2,t}^\top \end{bmatrix}^\top$ as

$$\begin{aligned} \min_{\eta_t} V(\eta_t) = \min_{\eta_t} \frac{1}{2} \Big\| & R\left(\tilde{q}_t^{GS_1} \right) \left(\mathcal{I}_3 + [\eta_{S_1,t} \times] \right) a_{jc,t}^{S_1} \\ & - R\left(q_t^{GS_2} \right) \left(\mathcal{I}_3 + [\eta_{S_2,t} \times] \right) a_{jc,t}^{S_2} \Big\|_2^2, \end{aligned} \tag{11}$$

where $\| \cdot \|_2$ denotes the two-norm. Inspired by [9], in (11), we use (3) but write the orientation in terms of a linearization point and an associated deviation instead of using unit-quaternions as

$$R\left(q_t^{GS_i} \right) = R\left(\tilde{q}_t^{GS_i} \right) \exp_{\mathrm{R}}\left(\eta_{S_i,t} \right) \tag{12a}$$

$$\begin{aligned} \exp_{\mathrm{R}}\left(\eta_{S_i,t} \right) &= \mathcal{I}_3 + \sin\alpha [v\times] + (1 - \cos\alpha)[v\times]^2 \\ &\approx \mathcal{I}_3 + [\eta_{S_i,t} \times] \end{aligned} \tag{12b}$$

where the approximation in (12b) assumes small $\eta_{S_i,t}$. By rewriting the problem in this fashion, we can optimize over an orientation deviation in terms of a rotation vector, rather than using a unit-quaternion. Therefore we avoid issues regarding quaternion normalization. As an additional consequence of using orientation deviation states instead of a unit-quaternion, the length of the optimization variables is reduced from four to three.

Like [10], [11] we solve the objective function of (11) in a single gradient descent iteration, instead of solving the problem for each time step. The benefit of doing a single gradient descent iteration, is the decrease in computational time and because of the high sampling frequency of the sensors, the corrections that need to be made are typically minor and the estimates will converge over time. Linearising $V(\eta_t)$ from (11) around $\tilde{q}_t^{GS_i} = \hat{q}_{t-1}^{GS_i}$ and $\eta_t = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$ using (12b), the gradient descent steps is given by

$$\hat{\eta}_t = -\mu_t \nabla V(\eta_t) \tag{13a}$$

$$\begin{aligned} \nabla V(\eta_t) = \Bigg(& -R\left(\hat{q}_{t-1}^{GS_1} \right) \begin{bmatrix} \left[a_{jc,t}^{S_1} \times \right] & 0_3 \end{bmatrix} \\ & + R\left(\hat{q}_{t-1}^{GS_2} \right) \begin{bmatrix} 0_3 & \left[a_{jc,t}^{S_2} \times \right] \end{bmatrix} \Bigg)^\top \\ & \left(R\left(\hat{q}_{t-1}^{GS_1} \right) a_{\mathrm{jc},t}^{S_1} - R\left(\hat{q}_{t-1}^{GS_2} \right) a_{\mathrm{jc},t}^{S_2} \right) \end{aligned} \tag{13b}$$

where μ_t is the gradient descent step length. The estimate η_t can subsequently be used to compute $q_{jc,t}^{GS_i}$ from (6) as

$$\begin{aligned} q_{jc,t}^{GS_i} &= q_{t-1}^{GS_i} \odot \exp_{\mathrm{q}} \left(\frac{1}{2} \hat{\eta}_{S_i,t} \right) \\ &\approx q_{t-1}^{GS_i} + \frac{1}{2} S\left(q_{t-1}^{GS_i} \right) \hat{\eta}_{S_i,t} \end{aligned} \tag{14}$$

C. Resulting algorithm

Inserting (14) and (10) in (6) as

$$\begin{aligned}\hat{q}_t^{GS_i} =& \hat{q}_{t-1}^{GS_i} + \frac{1}{2} S\left(\hat{q}_{t-1}^{GS_i}\right)\left(\gamma_t T y_{\omega,t}^{S_i}\right. \\ & \left. -\mu_t\left(1-\gamma_t\right)\nabla V\left(\eta_t\right)\right)\end{aligned} \tag{15}$$

The only thing that is left is choosing the values for γ_t and μ_t. When the estimation mainly relies on the integration of the gyroscope measurement and uses the accelerometer measurements to correct for integration drift one typically chooses $\gamma_t \approx 1$ [10], [11]. Here, the scaling factor for the gradient descent direction, $\mu_t\left(1-\gamma_t\right)$, is chosen to be $\frac{\beta T}{\|\nabla V(\eta_t)\|}$. This is done to enhance the algorithm's robustness against measurement model imperfections. The choice of β depends on the amount of drift that is to be expected from the gyroscope measurement integration. When the noise is assumed to be $e_{\omega,t} \sim \mathcal{N}\left(0, \sigma_\omega^2\right)$. Integration of the gyroscope measurements in one dimension results in an integration drift distributed as $Te_{\omega,t} \sim \mathcal{N}\left(0, T^2\sigma_\omega^2\right)$ [9]. It is known that the gyroscope measurement is a three-dimensional vector, the standard deviation of the integration drift for the unit quaternion is then given by $\sqrt{3}\sigma_\omega T$ [9]. This is a good choice for βT. The resulting filter equation with $\hat{q}_t^{GS_i}$ is as

$$\hat{q}_t^{GS_i} \approx \hat{q}_{t-1}^{GS_i} + \frac{T}{2} S\left(\hat{q}_{t-1}^{GS_i}\right)\hat{\omega}_t^{S_i}, \tag{16}$$

with

$$\hat{\omega}_t^{S_i} = y_{\omega,t}^{S_i} - \beta\frac{\nabla V\left(\eta_{S_i,t}\right)}{\|\nabla V\left(\eta_{S_i,t}\right)\|}. \tag{17}$$

Then the relative orientation can be computed as

$$\hat{q}_t^{S_1 S_2} = \left(\hat{q}_t^{GS_1}\right)^c \odot \hat{q}_t^{GS_2}, \tag{18}$$

where the superscript c denotes the unit-quaternion conjugate, the final algorithm for the orientation estimation by using a complementary filter is given in Algorithm 1. After this, (18) can be used to obtain the relative sensor orientation.

V. Numerical simulation

To validate our algorithm's workings, we perform several numerical simulations in which we can adapt and test the algorithm thoroughly. The data-set will contain the simulation of two distinct IMU. Both of these IMU have their angular velocities; to simulate the angular velocity measurement, we require a rotational speed. Here a the rotational speed of $\sin\left(\frac{\pi}{100T}t\right)$ rad/s will be assigned to IMU number one and $-\sin\left(\frac{\pi}{100T}t\right)$ rad/s to number two. This entails that both IMU rotates in the opposite direction of each other. This rotational speed is chosen, because the maximal rotational speed would then be one radiant per second, e.g., $\approx 57.3^\circ$ per second, which is in the range of human-like movement. One cycle will take 800 samples, e.g., 80 seconds since the sampling frequency is set equal to 10 Hz. The first 200 samples the simulated IMU will not rotate, after not rotating for 200 samples the IMU will start with rotating around the x-axis with the rotational speed as mentioned above, after this, the IMU will stop rotating in the x direction and move on by rotating for 200 samples in the y direction. The last 200 samples will be spent rotating around the z axis; after this, the cycle will be repeated until 8000 samples are reached, i.e., ten full cycles.

Algorithm 1 Sensor orientation estimation using an Complementary Filter with orientation deviation states

Input: Gyroscope & Accelerometer measurement $y_{\omega,t}^{S_i} y_{a,t}^{S_i}$, Initial Orientation $\hat{q}_1^{GS_i}$, distance from IMU to jc r_1 & r_2, tuning parameter β and sampling time T.
Output: Orientation estimate $\hat{q}_t^{GS_i}$.

1: Compute the gradient at time t

$$\begin{aligned}\nabla V\left(\eta_t\right) = & \left(-R\left(\hat{q}_{t-1}^{GS_1}\right)\left[\left[a_{jc,t}^{S_1}\times\right] \quad 0_3\right]\right. \\ & \left. +R\left(\hat{q}_{t-1}^{GS_2}\right)\left[0_3 \quad \left[a_{jc,t}^{S_2}\times\right]\right]\right)^\top \\ & \left(R\left(\hat{q}_{t-1}^{GS_1}\right) a_{\mathrm{jc},t}^{S_1} - R\left(\hat{q}_{t-1}^{GS_2}\right) a_{\mathrm{jc},t}^{S_2}\right)\end{aligned} \tag{19}$$

with $y_{a,t}^{S_i}$ and $\hat{q}_{t-1}^{GS_i}$

2: Obtain the updated orientation

$$\hat{q}_t^{GS_i} \approx \hat{q}_{t-1}^{GS_i} + \frac{T}{2} S\left(\hat{q}_{t-1}^{GS_i}\right)\hat{\omega}_t, \tag{20}$$

with $\hat{\omega}_t$ and $S\left(\hat{q}_{t-1}^{GS}\right)$ as defined in (17) and (9) respectively.

From this we can easily create a derivative which is the angular acceleration. For IMU number one this will be $\frac{\pi}{100T}\cos\left(\frac{\pi}{100T}t\right)$ rad/s^2 and $-\frac{\pi}{100T}\cos\left(\frac{\pi}{100T}t\right)$ rad/s^2 for the second IMU. The IMU also requires accelerometer measurements, these can be constructed by deconstructing (4). We first create the joint acceleration as a random signal between 10 and -10 and rotate it, such that the signal is in the first sensor frame. Together with the angular velocity, angular acceleration, and the distance to the jc we are able to create an acceleration signal for IMU one. This exact same deconstruction can be applied to the second IMU to obtain the acceleration signal for IMU two. The distances to the jc are chosen to be $r_1 = [1 \quad 0 \quad 0]^\top$ and $r_2 = [-1 \quad 0 \quad 0]^\top$. After this, simulated noise is added, where $e_{\omega,t} \sim \mathcal{N}\left(0, \Sigma_\omega\right)$ and $e_{a,t} \sim \mathcal{N}\left(0, \Sigma_a\right)$ with $\sigma_\omega = \frac{5\pi}{180}$ rad/s and $\sigma_a = -\frac{g}{100}$ m/s^2. During each simulation the angular acceleration is required for the calculation of the joint acceleration, this done via a five-point stencil [12].

A. Gaussian noise with known characteristics

Inspired by [9], our first simulation considers a situation where we have prior knowledge over the noise covariance matrices and initial orientation. The parameter β is chosen as explained in Section IV-C. As shown in Table I, desirable value for the Root Mean Squared Error (RMSE) in all three directions is obtained. This entails that the algorithm performs well when no additional disturbances are added.

B. Accelerometer model inaccuracies

In the real world, accelerometer outliers can be caused by the IMU or the body that the IMU is placed upon banging or ticking against a hard surface. An example of this would be that when the IMU is placed on the wrist of an arm, and the participant bumps into something with his/her arm. This will generate a spike in the accelerometer signal, which can be categorized as an accelerometer outlier. In the second simulation, we simulated these types of outliers to test the robustness of Algorithm 1. The outliers are created by replacing a percentage of the normalized accelerometer data with outliers. The outliers are chosen to be sampled from a Gaussian distribution with a covariance equal to an identity matrix. In this example, we choose to replace 5% of the normalized accelerometer data, as was similarly done in [9]. As shown in Table I, the outliers have little influence regarding the performance of the estimation algorithm.

C. Soft-Tissue Artifacts

Outliers that occur during banging or ticking against a hard surface are not the only type of disturbance that frequently occurs when performing evaluations that require the IMU to be worn on the human body. STA may occur due violations in the assumption that the sensor's position with respect to the jc is fixed at all times since the surface on which the sensors are placed can move to some degree. The STA can be simulated as an additive disturbance to the acceleration measurements as a linear combination of the angular acceleration measurements as [13]

$$d_a = H\dot{\omega} \tag{21}$$

Here the matrix H is a 3×3 matrix containing the STA model parameters. During real-world setups and testing, the elements of H will depend on multiple factors, i.e., sensor placement and the participant's physiology [13]. Similar to [13], we choose element of H to be randomized as a zero-mean Gaussian distribution, where the $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad, which equals 0.057 m/rad.

The maximal value for the STA, with a standard deviation as mentioned above, is around $4 \cdot 10^{-3}$ m/s^2 if we apply an acceleration of this magnitude to a sensor, then the displacement after 10^{-1} seconds would be roughly 40 μm, this is a rather small value but will serve as a good approximation for low values of simulated STA. As can be seen in Table I, the algorithm is able to reject the disturbances induced by low values of STA consistently. By changing the standard deviation used in our H matrix, we can simulate both moderate and high STA levels. A good approximation for moderate levels of STA is chosen to be $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad, which roughly translate to a maximum displacement of 0.4 cm after 10^{-1} seconds. For high levels of STA we choose a STA standard deviation of $\sigma_{d_a} = \frac{18}{\pi}$ m/rad, which translates to a maximum displacement of 4 cm after 10^{-1} seconds. Table I, shows that the estimation algorithm is robust for low and moderate values of STA, and is not capable of rejecting high levels of STA.

D. Wrong initial orientation

We want to see if the algorithm presented in Algorithm 1 can converge while a wrong initial orientation is given, and the measurements are subjected to noise, outliers, and moderate STA. The wrong initial orientation can be chosen arbitrarily, we chose a fixed wrong initial orientation as $\begin{bmatrix}81.87^\circ & -19.47^\circ & 135.00^\circ\end{bmatrix}$ for sensor one and $\begin{bmatrix}135.00^\circ & 19.47^\circ & 98.13^\circ\end{bmatrix}$ for sensor two. In Fig. 2 we show the estimation error for the first 200 samples of 100 Monte Carlo simulations with the mean and a shaded area, which corresponds to three standard deviations. This allows us to capture 99.73 $\%$ of all the data created during this simulation.

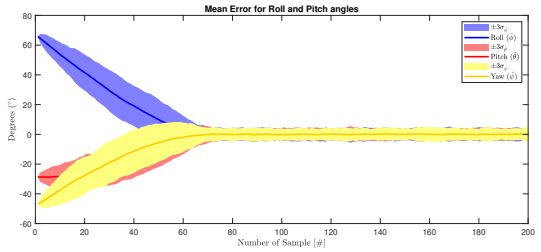


Fig. 2: Estimation error for the first 200 samples of 100 Monte Carlo simulations with a fixed wrong initial orientation as $81.87°$, $-19.47°$ and $135.00°$ for the roll, pitch and yaw of sensor one respectively and $135.00°$, $19.47°$ and $98.13°$ for the roll, pitch and yaw of sensor two respectively.

We observe an clear convergence towards zero, but once it has reached convergence around zero, the estimation error starts to fluctuate around zero. This is due to the noise, the accelerometer outliers and the STA that we have induced on the simulated IMU. From the combined results obtained above, we can conclude that the relative orientation estimation algorithm presented in Algorithm 1 is robust against noise, outliers, low and moderate levels of STA.

VI. Experimental Results

In this research project, we did not have access to an optical reference system due to the Covid-19 virus. Luckily, we were able to obtain optical reference data along with corresponding IMU data from MSc. I. Weygers from the KU Leuven, who is the author of [1]. We have acquired three different data-sets where two are regular gait analyses, and one data-set is of a participant performing a forward lunge movement [14]. The IMU data-sets were obtained via two MTw Awinda IMU from Xsens. The optical motion analysis reference system consisted of 13 infrared cameras (VICON Vero, Vicon Motion Systems Ltd). Both systems were measured at a sample rate of 100 Hz. Hardware time synchronization was used to capture inertial measurements and marker trajectories simultaneously. For the experimental results in [1], eleven healthy human participants with no history of knee surgery before the testing participated in the experiments conducted for [1]. For the experimental results in this section, it should be noted that it is unknown from

Simulation number		Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	Yaw ($\hat{\psi}$)
1	Known noise, variance and initial orientation	1.2968°	1.2922°	1.2927°
2	5% outliers magnitude $\mathcal{N}(0, \mathcal{I})$	1.4147°	1.4091°	1.4151°
3	Low STA $\sigma_{d_a} = \frac{0.018}{\pi}$ m/rad	1.4124°	1.4102°	1.4089°
4	Moderate STA $\sigma_{d_a} = \frac{1.8}{\pi}$ m/rad	1.5397°	1.5412°	1.5366°
5	High STA $\sigma_{d_a} = \frac{18}{\pi}$ m/rad	3.0062°	3.0056°	3.0065°

TABLE I: Effects on the RMSE of the relative sensor orientation with a complementary filter per simulation.

which of the eleven participants we obtained the data. For the first experiment, the IMU was attached latero-cranial on the shank and on the lateral side at mid-distance on the thigh via Velcro strips [1]. For the second and third experiments, the sensors are placed more anterior, i.e., to the front. This causes more movement of the sensor, and thus higher levels of STA occur during these trials. Special 3-D printed cases were created that contain reflective marker clusters, which could house the MTw Awinda sensor. This allowed for simultaneously capturing data from the IMU as well as data from the optical reference system. At the start of each experiment, the participants were asked not to move for five seconds. Subsequently, they were asked to perform either the forward lunge movement or a walk. During the first and second experiments, low and moderate STA levels are observed, while in the third, high levels of STA are observed, which is due to the movement of the muscle during the forward lunge movement and the placement of the sensors. To compare an orientation estimation obtained via sensor measurement $\left(\hat{q}_t^{GS_1}, \hat{q}_t^{GS_2}\right)$ to the orientation obtained via the optical reference system $\left(q_t^{OM_1}, q_t^{OM_2}\right)$, the constant misalignment $q^{M_2S_2}$, and $q^{M_1S_1}$ are estimated from the relative orientation references $q_t^{M_1M_2}$, and relative orientation estimates $\hat{q}_t^{S_1S_2}$, with the objective function, as

$$\begin{aligned} &q^{S_1M_1} \odot q_t^{M_1M_2} \odot q^{M_2S_2} \approx q_t^{S_1G} \odot q^{GO} \odot q^{OG} \odot q_t^{GS_2}, \\ &q_t^{M_1M_2} \odot q^{M_2S_2} \approx q^{M_1S_1} \odot q_t^{S_1S_2}. \end{aligned} \tag{22}$$

Together with the data-sets that were granted, we also obtained the results per data-set when the proposed Multiplicative Extended Kalman Filter (MEKF) method from [1] was used. This allows us to compare our proposed method versus the MEKF approach. In [1], the first five seconds are used to empirically define the noise variance to create the covariance matrices for the MEKF. The initial orientations were set to $[1 \quad 0 \quad 0 \quad 0]^\top$ and the initial process clearance matrix was set as a six by six identity matrix. The measurement noise covariance matrix and the covariance matrix for the error induced by the joint acceleration rotation were both set to be a three by three identity matrix. The values for r_1 and r_2 were estimated via the method presented in [15], which minimized the sum of squared errors. We have approximately used the first five seconds of each trial to obtain the gyroscope's covariance, from which we determine the values for β. We found $\beta = 1.72 \cdot 10^{-1}$, $\beta = 1.65 \cdot 10^{-2}$ and $\beta = 1.89 \cdot 10^{-2}$ for the first, second and third trial respectively. The values for r_1 and r_2 are estimated via [13], which minimizes the sum of absolute errors. A useful property of this type of objective function formulation is the robustness against outliers in the data-set. The robustness is imposed since large residuals are weighted by their absolute value and not by their squared value in the objective function. With these parameters, the results can be found in Table II. We observe that the MEKF from [1] and our proposed complementary filter perform about the same during the first experiment. It should be noted that when no STA would occur, it is to be expected that the MEKF from [1] would outperform our proposed complementary filter. Since MEKF is known to be more accurate in comparison to complementary filters. The RMSE values in three directions are shown in Table II. During the second trial, the sensors were placed more anterior before performing the gait analyses, which causes moderate STA levels to occur during this trial. We see in Table II that our proposed algorithm can reject the disturbance in the form of STA better than the proposed MEKF from [1], and thus performs better. In the third and last trial, which contains movement-data from a series of forward lunge motions, high STA levels are present. The high levels of STA are reflected in the results, as shown in Table II. We observe that both our proposed complementary filter and the MEKF from [1] cannot create a reliable relative sensor orientation estimate. The decrease in performance was expected if we consider the numerical simulations performed in Section V, here it was already shown that when high levels of STA would occur, the estimation of the relative sensor orientation would become unreliable.

Trial			Roll ($\hat{\phi}$)	Pitch ($\hat{\theta}$)	Yaw ($\hat{\psi}$)
1	Low STA	Algorithm 1	0.9722°	1.4793°	1.0895°
		MEKF from [1]	1.5582°	1.4667°	1.1852°
2	Moderate STA	Algorithm 1	1.2337°	2.5043°	2.7399°
		MEKF from [1]	5.3134°	5.0609°	4.9964°
3	High STA	Algorithm 1	5.5601°	9.9807°	6.9118°
		MEKF from [1]	6.2792°	9.6570°	3.3983°

TABLE II: Results for Algorithm 1 versus the proposed MEKF that is presented in [1]. We observe a difference in performance for both methods. When low or moderate STA levels occur, our proposed method remains accurate, while the proposed MEKF from [1] becomes unreliable due to the disturbances caused by STA.

We observe a clear degradation regarding the performance of Algorithm 1, which is due to the increase in the levels of STA experienced during each trial.

VII. CONCLUSIONS

By using the constraints imposed by the human body itself in the form of rigidly connected segments, accompanied by a complementary filter. We were able to create an accurate estimation for the relative sensor orientation between the two sensors placed on adjacent segments. We have proven that this method is robust against high levels of gyroscope noise, accelerometer outliers, and low and moderate STA levels. We observed that when high levels of STA occur, the estimation algorithm is not able to perform a correct estimation anymore and thus becomes unreliable. Therefore, this could be a topic for further research in the near future.

ACKNOWLEDGMENT

REFERENCES

[1] I. Weygers, M. Kok, H. De Vroey, T. Verbeerst, M. Versteyhe, H. Hallez, and K. Claeys, "Drift-free inertial sensor-based joint kinematics for long-term arbitrary movements," *IEEE Sensors Journal*, vol. 20, no. 14, pp. 7969–7979, 2020.

[2] M. Noorkõiv, H. Rodgers, and C. I. Price, "Accelerometer measurement of upper extremity movement after stroke: a systematic review of clinical studies," *Journal of neuroengineering and rehabilitation*, vol. 11, no. 1, p. 144, 2014.

[3] S. G. Trost, K. L. Mciver, and R. R. Pate, "Conducting accelerometer-based activity assessments in field-based research," *Medicine & science in sports & exercise*, vol. 37, no. 11, pp. S531–S543, 2005.

[4] A. Chadwell, L. Kenney, M. Granat, S. Thies, J. S. Head, and A. Galpin, "Visualisation of upper limb activity using spirals: A new approach to the assessment of daily prosthesis usage," *Prosthetics and orthotics international*, vol. 42, no. 1, pp. 37–44, 2018.

[5] M. Kok, J. D. Hol, and T. B. Schön, "Using inertial sensors for position and orientation estimation," *Foundations and Trends in Signal Processing*, vol. 11, no. 1-2, pp. 1–153, 2017.

[6] ——, "An optimization-based approach to human body motion capture using inertial sensors," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 79–85, 2014.

[7] W. Teufl, M. Miezal, B. Taetz, M. Fröhlich, and G. Bleser, "Validity, test-retest reliability and long-term stability of magnetometer free inertial sensor based 3d joint kinematics," *Sensors*, vol. 18, no. 7, p. 1980, 2018.

[8] A. Leardini, L. Chiari, U. Della Croce, and A. Cappozzo, "Human movement analysis using stereophotogrammetry: Part 3. soft tissue artifact assessment and compensation," *Gait & posture*, vol. 21, no. 2, pp. 212–225, 2005.

[9] M. Kok and T. B. Schön, "A fast and robust algorithm for orientation estimation using inertial sensors," *IEEE Signal Processing Letters*, vol. 26, no. 11, pp. 1673–1677, 2019.

[10] S. Madgwick, "An efficient orientation filter for inertial and inertial/magnetic sensor arrays," *Report x-io and University of Bristol (UK)*, vol. 25, pp. 113–118, 2010.

[11] S. O. Madgwick, A. J. Harrison, and R. Vaidyanathan, "Estimation of imu and marg orientation using a gradient descent algorithm," in *2011 IEEE international conference on rehabilitation robotics*. IEEE, 2011, pp. 1–7.

[12] T. Sauer, *Numerical Analysis*. Pearson, 2012, vol. 2.

[13] F. Olsson and K. Halvorsen, "Experimental evaluation of joint position estimation using inertial sensors," in *2017 20th International Conference on Information Fusion (Fusion)*. IEEE, 2017, pp. 1–8.

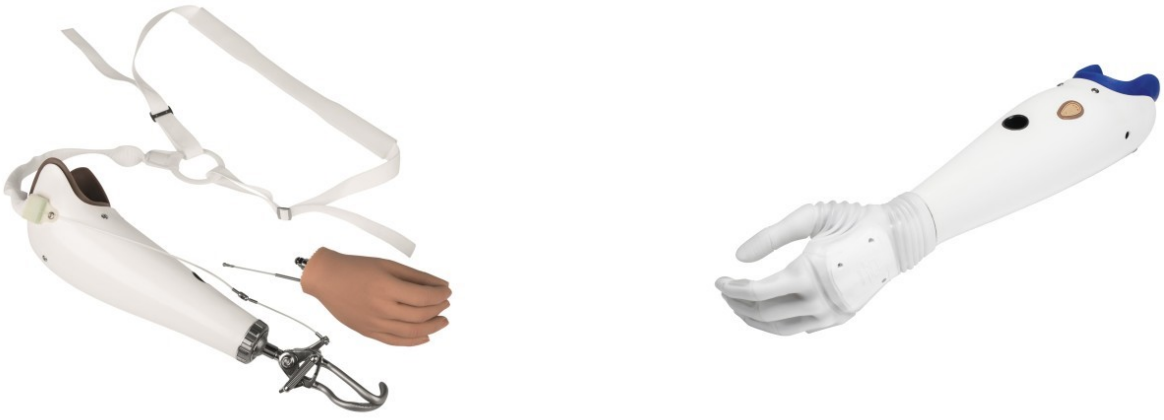
[14] H. De Vroey, F. Staes, I. Weygers, E. Vereecke, G. Van Damme, H. Hallez, and K. Claeys, "Hip and knee kinematics of the forward lunge one year after unicondylar and total knee arthroplasty," *Journal of Electromyography and Kinesiology*, vol. 48, pp. 24–30, 2019.

[15] T. Seel, T. Schauer, and J. Raisch, "Joint axis and position estimation from inertial measurement data by exploiting kinematic constraints," in *2012 IEEE International Conference on Control Applications*. IEEE, 2012, pp. 45–49.

Appendix B

Different Prostheses and the Self-Grasping Hand

There are currently several types of prostheses available; active, passive, and hybrid (combination between active and passive) prostheses [38]. The active prostheses can be divided into two main categories, myoelectric prostheses, and a body powered (BP) prostheses [53], which can be seen in Figure B-1. The first is a prosthetic limb controlled by the rapidly rising and falling of the membrane potential of the muscle membrane that is developed because of the excitation of the central nervous system. A myoelectric prosthesis is thus based on the principle that motions performed by a limb are preceded by currents in the muscle of that specific limb [25]. The latter is an artificial limb whereby shoulder movement and movements from the arm operate the device utilizing a shoulder harness and cable [37].



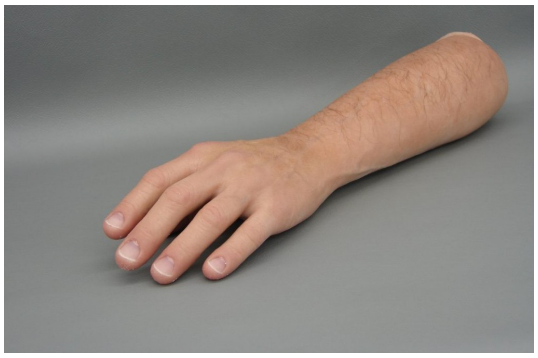
(a) BP prosthetic with hook and hand [42].

(b) The Michelangelo prosthetic hand [43].

Figure B-1: Two types of active prostheses; BP prosthesis and myoelectric prosthesis.

A passive prosthesis allows for no active movement of any of the joints and is considered

the lightest device because they contain no motors and few mechanical systems [38]. Passive prostheses can be divided into two categories, they can either be static or adjustable [30], and are shown in Figure B-2. The static prostheses can not be moved at all, while the adjustable prostheses feature an adjustable grasping mechanism or parts that can be adjusted to multiple orientations, these adjustments are performed by the non-artificial upper limb [30]. Although limited in their functionality, the passive prostheses are popular, since one out of three people with a limb deficiency uses a passive type prosthesis [30].



(a) High-definition nonadjustable silicone cosmetic upper limb prosthetic [47]

(b) A passive adjustable upper limb prosthesis [12] from the 19-th century.

Figure B-2: Two types of passive prostheses; static prosthesis and adjustable prosthesis.

Although there is a wide variety in available prostheses, rejection rates of upper limb prostheses are high [6]. The percentage of amputees that do not actively wear and use his/her device is estimated to be around 27% and the percentage of people that completely rejected the use of an artificial limb is 20% [53, 6]. The cause of these high rejection rates stems from different problems related to prostheses; e.g. pricing, life-like appearance, grip control, and weight [7]. Users of passive prosthetic hands have indicated that they experience functional difficulties with the performance of pulling, gripping, holding, and releasing objects in activities of daily living activities of daily living (ADLs) [30]. The static prostheses can hardly be improved regarding functionality. Adjustable prostheses, however, provide a more fruitful perspective when it comes to improvements. The first steps for improvement have already been taken. In

(a) The Self Grasping Hand (SGH) with cosmetic glove [1]

(b) The SGH without cosmetic glove [36]

Figure B-3: The SGH with and without cosmetic glove.

[54] the SGH is presented, which can be seen in Figure B-3. The SGH is a passive prosthetic hand that has articulating fingers and can perform the hook grip, power grip and pinch grip. The gripping of objects is controlled via indirectly pushing an object to the hand, or directly by pushing the prosthetic thumb against a fixed object [54]. The grip force of the SGH is proportional to the applied push force applied by the amputee. When the amputee stops exerting a push force, the grip force is locked and the object is being clutched. To release the object, a button, which can be seen in Figure B-3b, has to be pushed after which the object can be released by pushing the object slightly into the hand [54]. In recent research the SGH was compared to other passive adjustable prostheses, it was found that activities were performed 11% faster and required less user effort with the SGH and that the grasping function of the SGH was used 54% more often [54].

Appendix C

Parametizations

To understand the fundamentals of inertial Sensor Fusion (SF), it is first necessary to elaborate on the basics required to compute an orientation estimate successfully. Rotation matrices, Euler angles, axis-angle representation, and unit quaternions are all key concepts for representing rotations. Some of these parameterizations are not as intuitive in their use as others. Therefore, a summary is given in this chapter. The different parameterizations and their descriptions are ordered as follows. First, the rotation matrix description is given; after this, the Euler Angles are discussed, next to the axis-angle representation, and last, the unit quaternions are presented. After this, we will discuss the interchangeable use of the parameterizations and linearization. As of last, we will discuss Denavit-Hartenberg (DH) transformation that is required for a kinematic chain model.

C-1 Rotation Matrices

In this study, only rotations in $\mathbb{R}^3$ are considered, and hence we only work with the special orthogonal group $SO(3)$. If we were to consider a rotation matrix $R \in \mathbb{R}^{3\times 3}$, it will entail the following properties as specified in Eq. (C-4). To explain what a $SO(3)$ is, it is first necessary to correctly explain what a General Linear group is and what an Orthogonal group is.

If we consider the pair $(\mathcal{M}^{n\times n}, \cdot)$, where $\mathcal{M}^{n\times n}$ is defined as the set of all invertible matrices and the product $\cdot$ is the regular matrix product. Then $GL(\mathbb{R})_n = (\mathcal{M}^{n\times n}, \cdot)$ is a group and is referred to as an General Linear group. For the general linear group, let $\mathcal{M}$ be a 2×2 matrix as

$$\mathcal{M}^{2\times 2} = \left\{(a_{11}, a_{12}, a_{21}, a_{22}) \in \mathbb{R}^4 : a_{11}a_{22} - a_{12}a_{21} \neq 0\right\} \tag{C-1}$$

and let the product be defined as

$$\begin{aligned}\phi\left((a_{11}, a_{12}, a_{21}, a_{22}), (b_{11}, b_{12}, b_{21}, b_{22})\right) &:= \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \\ &= (a_{11}b_{11} + a_{12}b_{21}, \cdots).\end{aligned} \tag{C-2}$$

If we now again consider the rotation matrix R, The General Linear group can mathematically be expressed as follows

$$GL(\mathbb{R})_3 = \left\{ R \in \mathcal{M}^{3\times 3}(\mathbb{R}) : \det(R) \neq 0 \right\}. \tag{C-3a}$$

For a matrix to be part of the Orthogonal group $O(3)$ an additional property is required, a 3×3 orthogonal matrix has the property that $RR^T = R^T R = I_3$

$$O(3) = \left\{ R \in GL(\mathbb{R})_3 : R^\top R = I_3 \right\}, \tag{C-3b}$$

The special orthogonal group has an additional property of the determinant being one as

$$SO(3) = \{R \in O(3) : \det(R) = 1\}, \tag{C-4}$$

Considering the rotation matrix R and the two coordinate frames, U and V frame. A vector x expressed in the V frame can be rotated to the U frame as

$$x^U = R^{UV} x^V, \tag{C-5a}$$

while going from the U to the V frame can be done by

$$x^V = (R^{UV})^T x^U. \tag{C-5b}$$

C-2 Euler Angles

Any rotation of a 3D body can be represented as three consecutive rotations around three orthogonal axes. Representing a vector rotation in this way makes use of so-called Euler angles. Each of these rotations is a simple extension of a 2D rotation matrix since it performs a 2D rotation with respect to two axes while leaving the coordinate of the third axis unchanged [26]. In this study, we adopt the convention that the rotations are performed in a specific order; we choose to rotate around the z-axis, then rotate around the y-axis, and last, we rotate around the x-axis. These rotations around the different axis all have a specific name.

1. Yaw is defined as a rotation of ψ about the z-axis. This generates the following rotation matrix

$$R_z(\psi) = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{C-6}$$

2. Pitch is defined as a rotation of θ about the y-axis. This generates the following rotation matrix

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}. \tag{C-7}$$

3. Roll is defined as a rotation of ϕ about the x-axis. This generates the following rotation matrix

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}. \tag{C-8}$$

If we now consider that a 3D body in the V frame is rotated with respect to the U frame by (ψ, θ, ϕ), we have to multiply the rotation matrices in the correct order to obtain the complete rotation matrices.

$$R^{UV} = R^{UV}(\phi)R^{UV}(\theta)R^{UV}(\psi) \tag{C-9}$$

By substituting Eq. (C-9) with Eq. (C-6), Eq. (C-7) and Eq. (C-8) the following rotation matrix is obtained

$$R^{UV} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix}. \tag{C-10}$$

A problem with Euler angles is that they are non-unique, this is due to wrapping and gimbal lock [23]. Wrapping is the effect that occurs when a rotation is "wrapped" around its own axis. This can only occur when $2\pi k$ is added to the angle of rotation. If we look at a rotation of $(\frac{\pi}{2}, 0, 0)$, we can easily observe that $(\frac{\pi}{2} + 2\pi k, 0, 0)$, where k can only be positive or negative integers, will give the exact same rotation.

$$\begin{aligned} R^{GS} &= \begin{bmatrix} \cos 0\cos\frac{\pi}{2} & \cos 0\sin\frac{\pi}{2} & -\sin 0 \\ \sin 0\sin 0\cos\frac{\pi}{2} - \cos 0\sin\frac{\pi}{2} & \sin 0\sin 0\sin\frac{\pi}{2} + \cos 0\cos\frac{\pi}{2} & \sin 0\cos 0 \\ \cos 0\sin 0\cos\frac{\pi}{2} + \sin 0\sin\frac{\pi}{2} & \cos 0\sin 0\sin\frac{\pi}{2} - \sin 0\cos\frac{\pi}{2} & \cos 0\cos 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{C-11}$$

Gimbal lock is the effect that occurs when two of the three gimbals end up in a parallel arrangement. This creates a loss of one degree of freedom in a three-dimensional system. For example, the rotations $(\frac{\pi}{2}, \frac{\pi}{2}, 0)$ and $(\pi, \frac{\pi}{2}, \frac{\pi}{2})$ will both give the same rotation matrix.

$$
\begin{aligned}
R^{GS} &= \begin{bmatrix} \cos\frac{\pi}{2}\cos\frac{\pi}{2} & \cos\frac{\pi}{2}\sin\frac{\pi}{2} & -\sin\frac{\pi}{2} \\ \sin 0\sin\frac{\pi}{2}\cos\frac{\pi}{2} - \cos 0\sin\frac{\pi}{2} & \sin 0\sin\frac{\pi}{2}\sin\frac{\pi}{2} + \cos 0\cos\frac{\pi}{2} & \sin 0\cos\frac{\pi}{2} \\ \cos 0\sin\frac{\pi}{2}\cos\frac{\pi}{2} + \sin 0\sin\frac{\pi}{2} & \cos 0\sin\frac{\pi}{2}\sin\frac{\pi}{2} - \sin 0\cos\frac{\pi}{2} & \cos 0\cos\frac{\pi}{2} \end{bmatrix} \\
&= \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} \cos\frac{\pi}{2}\cos\pi & \cos\frac{\pi}{2}\sin\pi & -\sin\frac{\pi}{2} \\ \sin\frac{\pi}{2}\sin\frac{\pi}{2}\cos\pi - \cos\frac{\pi}{2}\sin\pi & \sin\frac{\pi}{2}\sin\frac{\pi}{2}\sin\pi + \cos\frac{\pi}{2}\cos\pi & \sin\frac{\pi}{2}\cos\frac{\pi}{2} \\ \cos\frac{\pi}{2}\sin\frac{\pi}{2}\cos\pi + \sin\frac{\pi}{2}\sin\pi & \cos\frac{\pi}{2}\sin\frac{\pi}{2}\sin\pi - \sin\frac{\pi}{2}\cos\pi & \cos\frac{\pi}{2}\cos\frac{\pi}{2} \end{bmatrix}
\end{aligned} \tag{C-12}
$$

C-3 Axis-Angle Representation

Another representation for a rotation is via axis-angle representation. The axis-angle representation allows us to represent a rotation between two coordinate frames in terms of an angle α and a unit vector n.

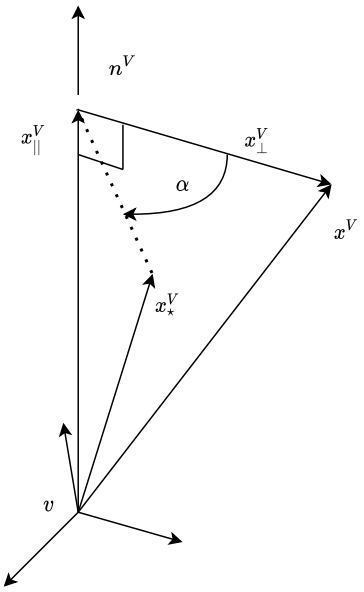


Figure C-1: A clockwise rotation of vector x by an angle α around the unit vector n [23].

For the explanation of the axis-angle representation, we adopt the same notation as used in [23]. In Figure C-1, vector x^V is shown which is rotated by an angle α around a vector of unit length n^V. The vector $x_{\star}^V$ represents the final vector after it is rotated by the angle α.

From Figure C-1, we can already conclude that x^V can be decomposed into the orthogonal and parallel component.

$$x^V = x_{\parallel}^V + x_{\perp}^V \tag{C-13a}$$

The parallel component $x_{\parallel}^{V}$ can be written as

$$x_{\parallel}^{V} = \left(x^{V} \cdot n^{V}\right) n^{V}, \tag{C-13b}$$

which is due to the property of the inner product that allows us to write a inner product as follows [49]

$$\left(x^{V} \cdot n^{V}\right) = |x^{V}||n^{V}| \cos\beta. \tag{C-13c}$$

Here β is the angle between the two vectors of which the inner product is taken (in this case x^{V} and n^{V}). The same decomposition as found in Eq. (C-13a) can be applied to $x_{\star}^{V}$. by doing so, we obtain

$$x_{\star}^{V} = \left(x_{\star}^{V}\right)_{\parallel} + \left(x_{\star}^{V}\right)_{\perp}, \tag{C-14}$$

obviously both the parallel decomposition's are equal to one another

$$\left(x_{\star}^{V}\right)_{\parallel} = x_{\parallel}^{V}, \tag{C-15a}$$

and the perpendicular decomposition can be written as

$$\left(x_{\star}^{V}\right)_{\perp} = x_{\perp}^{V} \cos\alpha + \left(x^{V} \times n^{V}\right) \sin\alpha, \tag{C-15b}$$

Since, $\left(x_{\star}^{V}\right)_{\parallel}$ and $\left(x_{\star}^{V}\right)_{\perp}$ can be expressed in terms of x^{V}, α and n^{V}, it is possible to fill in Eq. (C-14), by doing so the following expression is obtained

$$\begin{aligned} x_{\star}^{V} &= \left(x^{V} \cdot n^{V}\right) n^{V} + \left(x^{V} - \left(x^{V} \cdot n^{V}\right) n^{V}\right) \cos\alpha + \left(x^{V} \times n^{V}\right) \sin\alpha \\ &= x^{V} \cos\alpha + n^{V} \left(x^{V} \cdot n^{V}\right) (1 - \cos\alpha) - \left(n^{V} \times x^{V}\right) \sin\alpha. \end{aligned} \tag{C-16}$$

If we introduce a coordinate frame U, where $x_{\star}^{V}$ is equivalent to x^{U} and substitute x^{U} for $x_{\star}^{V}$ in Eq. (C-16) we get

$$x^{U} = x^{V} \cos\alpha + n^{V} \left(x^{V} \cdot n^{V}\right) (1 - \cos\alpha) - \left(n^{V} \times x^{V}\right) \sin\alpha. \tag{C-17}$$

It should be noted that the combination of the unit vector n and the rotation angle α, or $\eta = n\alpha$, indicates the rotation vector or the axis-angle parameterization. We now show the equivalence between the rotation matrix parameterization and the axis-angle parameterization. Before doing so it is needed to shed some light on several properties of the cross product. Given the vectors u and v we have [23],

$$v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}, \quad v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}, \tag{C-18}$$

$$u \times v = [u\times]v = -[v\times]u, \quad [u\times] \triangleq \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}. \tag{C-19}$$

Moreover, multiple cross products between vectors can be expanded in terms of the inner product. Let us introduce a vector w, which is also a 3×1 vector and hence structurally equivalent to the vectors u and v then

$$u \times (v \times w) = v(w \cdot u) - w(u \cdot v). \tag{C-20}$$

By utilizing these relations, Eq. (C-17) can be transformed into the following

$$\begin{aligned} x^U &= x^V \cos\alpha + n^V \left(x^V \cdot n^V\right)(1-\cos\alpha) - \left(n^V \times x^V\right)\sin\alpha \\ &= x^V \cos\alpha + \left(n^V \times \left(n^V \times x^V\right) + x^V\right)(1-\cos\alpha) - \left(n^V \times x^V\right)\sin\alpha \\ &= \left(\mathcal{I}_3 - \sin\alpha \left[n^V\times\right] + (1-\cos\alpha)\left[n^V\times\right]^2\right)x^V. \end{aligned} \tag{C-21}$$

If we look at Eq. (C-5a) and Eq. (C-21), the rotation matrix parameterization can be written as

$$R^{UV}\left(n^V, \alpha\right) = \mathcal{I}_3 - \sin\alpha \left[n^V\times\right] + (1-\cos\alpha)\left[n^V\times\right]^2. \tag{C-22}$$

It is worth noticing that the rotation matrix, $R^{UV}\left(n^V, \alpha\right)$, can also be written as

$$R^{UV}\left(n^V, \alpha\right) = \exp\left(-\alpha\left[n^V\times\right]\right). \tag{C-23}$$

Like Euler Angles, this way of parameterization is prone to wrapping.

C-4 Unit Quaternions

The unit quaternion is a 4-dimensional parameterization q, which has its length equal to 1. Hence the word "unit" in "unit quaternion".

$$q = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^\top = \begin{bmatrix} q_0 \\ q_v \end{bmatrix}, \quad q \in \mathbb{R}^4, \quad \|q\|_2 = 1 \tag{C-24}$$

Describing a rotation using the unit quaternion parameterization can be done as follows

$$\bar{x}^U = q^{UV} \odot \bar{x}^V \odot \left(q^{UV}\right)^c, \tag{C-25}$$

here the conjugate of the quaternion is represented as

$$q^c = \begin{bmatrix} q_0 & -q_v^\top \end{bmatrix}^\top, \tag{C-26}$$

and

$$\bar{x}^V = \begin{bmatrix} 0 & \left(x^V\right)^\top \end{bmatrix}^\top \quad . \tag{C-27}$$

The quaternion multiplication operator is denoted as $\odot$, and represents the following operation

$$p \odot q = \begin{bmatrix} p_0 q_0 - p_v \cdot q_v \\ p_0 q_v + q_0 p_v + p_v \times q_v \end{bmatrix} = p^L q = q^R p. \tag{C-28}$$

The superscripts L and R denote the left side multiplication and right side multiplication respectively and are denoted as

$$p^L \triangleq \begin{bmatrix} p_0 & -p_v^\top \\ p_v & p_0 \mathcal{I}_3 + [p_v \times] \end{bmatrix}, \quad q^R \triangleq \begin{bmatrix} q_0 & -q_v^\top \\ q_v & q_0 \mathcal{I}_3 - [q_v \times] \end{bmatrix}. \tag{C-29}$$

Some additional important definitions with the left- and right hand side multiplication are

$$p^L q^R = q^R p^L \tag{C-30}$$

and

$$(q^c)^L = \left(q^L\right)^\top, \quad (q^c)^R = \left(q^R\right)^\top \tag{C-31}$$

If we now fill in Eq. (C-25) with the left- and right hand side multiplication as defined in Eq. (C-29), $\bar{x}^U$ becomes

$$\begin{aligned} \bar{x}^U &= \left(q^{UV}\right)^L (q^{vu})^R \bar{x}^V \\ &= \begin{bmatrix} q_0 & -q_v^\top \\ q_v & q_0 \mathcal{I}_3 + [q_v \times] \end{bmatrix} \begin{bmatrix} q_0 & q_v^\top \\ -q_v & q_0 \mathcal{I}_3 + [q_v \times] \end{bmatrix} \begin{bmatrix} 0 \\ x^V \end{bmatrix} \\ &= \begin{bmatrix} q_0^2 + q_v^\top q_v & q_0 q_v^\top - q_0 q_v^\top - q_v^\top [q_v \times] \\ q_v q_0 - q_0 q_v - [q_v \times] q_v & q_v q_v^\top + q_0^2 \mathcal{I}_3 + 2 q_0 [q_v \times] + [q_v \times]^2 \end{bmatrix} \begin{bmatrix} 0 \\ x^V \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0_{1\times 3} \\ 0_{3\times 1} & q_v q_v^\top + q_0^2 \mathcal{I}_3 + 2 q_0 [q_v \times] + [q_v \times]^2 \end{bmatrix} \begin{bmatrix} 0 \\ x^V \end{bmatrix}. \end{aligned} \tag{C-32}$$

By looking back at Eq. (C-5a) we can conclude that a rotation matrix can be written as $R^{UV} = q_v q_v^\top + q_0^2 \mathcal{I}_3 + 2 q_0 [q_v \times] + [q_v \times]^2$. If we choose $q^{UV}\left(n^V, \alpha\right)$ to be as follows

$$q^{UV}\left(n^V, \alpha\right) = \begin{bmatrix} \cos \frac{\alpha}{2} \\ -n^V \sin \frac{\alpha}{2} \end{bmatrix}, \tag{C-33}$$

and substituting q_0 and q_v by those values in Eq. (C-32) we are able to obtain

$$
\begin{aligned}
\bar{x}^U &= \begin{bmatrix} 1 & 0_{1\times 3} \\ 0_{3\times 1} & (\mathcal{I}_3 + [n^V\times]^2)\sin^2\frac{\alpha}{2} + \cos^2\frac{\alpha}{2} - 2\sin\frac{\alpha}{2}\cos\frac{\alpha}{2}[n^V\times] + \sin^2\frac{\alpha}{2}[n^V\times] \end{bmatrix} \begin{pmatrix} 0 \\ x^V \end{pmatrix} \\
&= \begin{bmatrix} 1 & 0_{1\times 3} \\ 0_{3\times 1} & \mathcal{I}_3 - 2\cos\frac{\alpha}{2}\sin\frac{\alpha}{2}\left[n^V\times\right] + 2\sin^2\frac{\alpha}{2}\left[n^V\times\right]^2 \end{bmatrix} \begin{bmatrix} 0 \\ x^V \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0_{1\times 3} \\ 0_{3\times 1} & \mathcal{I}_3 - \sin\alpha\left[n^V\times\right] + (1-\cos\alpha)\left[n^V\times\right]^2 \end{bmatrix} \begin{bmatrix} 0 \\ x^V \end{bmatrix}.
\end{aligned}
\tag{C-34}
$$

C-5 Interchangeable Use of Parameterizations and Linearization

The four different parameterizations that have been discussed are all different in use. They all bring positive as well as negative properties when utilizing them as parameterization. These properties are summed up in the table below where we discuss whether they can represent all the attitudes (Global), if for each SO(3) attitude only one representation exists (Unique) and if the rotation parameterization is defined by 3 elements (Minimal).

Parameterization	Global	Unique	Minimal
Rotation Matrix	Yes	Yes	No-9 parameters
Euler Angles	No	No	Yes-3 parameters
Axis Angle	Yes	No	Yes-3 parameters
Quaternion	Yes	No	No-4 parameters

Table C-1: Summary of the four different parameterizations that have been discussed in Appendix C.

The parameterization one chooses to work with has a significant impact on the workings of the algorithm. Most estimation algorithms assume that the unknown states and parameters are represented in Euclidean space. For most of the parameterizations that have been discussed previously, this does not hold [23]. Next, a method is discussed that brings forth a solution to this problem. The group of three-dimensional rotations is referred to as special orthogonal group SO(3). This SO(3) group is characterized by several constraints and is a so-called matrix Lie group. A Lie group is a topological group that is also a smooth manifold. With every Lie group, a Lie algebra is associated, a Lie algebra is a vector space, which is the tangent space around the group's identity element . For more detailed information on Lie groups and Lie algebra, please refer to [18]. Because rotation is a matrix Lie group, there exists an exponential map. Utilizing this information gives rise to the possibility of representing orientations on SO(3) using unit quaternions or rotation matrices, while the orientation deviation parameterization is represented by using a three-dimensional rotation vector. As a result, an orientation q_t^{GS} can be encoded in terms of a linearization point, which is parameterized as a unit quaternion or rotation matrix denoted by $\tilde{q}_t^{GS}$ and $\tilde{R}_t^{GS}$ respectively. An orientation deviation can be indicated using a rotation vector of η_t.

$$q_t^{GS} = \exp\left(\frac{\bar{\eta}_t^G}{2}\right) \odot \tilde{q}_t^{GS}, \quad R_t^{GS} = \exp\left(\left[\eta_t^G \times\right]\right) \tilde{R}_t^{GS} \tag{C-35}$$

and for the quaternion exp operator and Eq. (C-23)

$$\exp(\bar{\eta}) = \begin{pmatrix} \cos\|\eta\|_2, \\ \|\eta\|_2 \sin\|\eta\|_2 \end{pmatrix} \tag{C-36a}$$

$$\exp([\eta\times]) = \mathcal{I}_3 + \sin\left(\|\eta\|_2\right)\left[\frac{\eta}{\|\eta\|_2}\times\right] + \left(1-\cos\left(\|\eta\|_2\right)\right)\left[\frac{\eta}{\|\eta\|_2}\times\right]^2. \tag{C-36b}$$

We adopt the same mappings as in [23] for notational convenience

$$q = \exp_{\mathrm{q}}(\eta), \quad \exp_{\mathrm{q}} : \mathbb{R}^3 \to \left\{q \in \mathbb{R}^4 : \|q\|_2 = 1\right\} \tag{C-37a}$$

$$R = \exp_{\mathrm{R}}(\eta), \exp_{\mathrm{R}} : \mathbb{R}^3 \to \left\{R \in \mathbb{R}^{3\times 3} : RR^T = \mathcal{I}_3, \det R = 1\right\}. \tag{C-37b}$$

This enables us to write Eq. (C-35)

$$q_t^{GS} = \exp_{\mathrm{q}}\left(\frac{\eta_t^G}{2}\right) \odot \tilde{q}_t^{GS}, \quad R_t^{GS} = \exp_{\mathrm{R}}\left(\eta_t^n\right) \tilde{R}_t^{GS}, \tag{C-38}$$

where the reverse mappings are defined as follows

$$\eta = \log_q(q) = \frac{\arccos q_0}{\sin\arccos q_0} q_v, \tag{C-39}$$

with $\log_{\mathfrak{q}} : \{q \in \mathbb{R}^4 : \|q\|_2 = 1\} \to \mathbb{R}^3$,

$$\eta = \log_{\mathrm{R}}(R) = \begin{pmatrix} (\log R)_{32} \\ (\log R)_{13} \\ (\log R)_{21} \end{pmatrix}, \tag{C-40}$$

with $\log_{\mathrm{R}} : \left\{R \in \mathbb{R}^{3\times 3} : RR^T = \mathcal{I}_3, \det R = 1\right\} \to \mathbb{R}^3$. Here $\log R$ represents the natural logarithm. It is also possible to make use of the following approximations. It should be noted that for these approximations it is required that η_t^G is small.

$$\begin{aligned} \exp_{\mathrm{q}}(\eta) &\approx \begin{pmatrix} 1 \\ \eta \end{pmatrix} \\ \log_{\mathfrak{q}}(q) &\approx q_v \\ \exp_{\mathrm{R}}(\eta) &\approx \mathcal{I}_3 + [\eta\times] \\ \log_{\mathrm{R}}(R) &\approx \begin{pmatrix} R_{32} & R_{13} & R_{21} \end{pmatrix}^\top \end{aligned} \tag{C-41}$$

C-6 Denavit Hartenberg Transformation

A DH transformation is composed of a rotation around the x-axis (angle ϕ), followed by two translations along the x- and the z-axis (A, D) and, finally, a rotation around the z-axis (angle ψ). This results in the following homogeneous transformation:

$$DH(\psi, D, A, \phi) = \begin{pmatrix} \cos\psi & -\sin\psi\cos\phi & \sin\psi\sin\phi & A\cos\psi \\ \sin\psi & \cos\psi\cos\phi & -\cos\psi\sin\phi & A\sin\psi \\ 0 & \sin\phi & \cos\phi & d \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{C-42}$$

These transformations are composed to a tree of paths:

$$DH_{path}\left(\{\psi_i, D_i, A_i, \phi_i\}_{i=1}^{n}\right) := \prod_{i=1}^{G} H_i = \prod_{i=1}^{n} DH\left(A_i, D_i, \phi_i, \psi_i\right). \tag{C-43}$$

A kinematic chain is constructed by having an initial transformation $H^{G\bar{W}}$, where $\bar{W}$ refers to the local world frame, i.e., the chain root. From there, paths lead to each Inertial Measurement Unit (IMU). Through this structure, each transformation is represented in the reference frame of its predecessor. Hence, the pose of the kth IMU in the global frame is given by:

$$H_k^{GS} = H^{G\bar{W}} \prod_{i=1}^{k} \left(H_i\right) H_k^{BS}. \tag{C-44}$$

According to the DH convention, per transformation, only one parameter may be variable. In our case, i.e., for human motion capturing with given segment lengths and sensor to segment calibrations, only (joint) angles are estimation variables. A chain state, x_c, (i.e., a state-space representing a kinematic chain) is, therefore, a set of variable angles. Consequently, the kth IMU pose in the global frame depends on the angles to its path:

$$H_k^{GS} = f_k\left(x_c\right), \tag{C-45}$$

where the right-hand side is given by Eq. (C-44). Note, $H^{G\bar{W}}$ is here assumed given and fixed.

Bibliography

[1] webredactie 3ME. *Gerwin Smit nominated biggest scientific talent 2018*. URL: https://www.tudelft.nl/en/2018/3me/april/gerwin-smit-nominated-biggest-scientific-talent-2018/. (accessed: 18.11.2019).

[2] ActiGraph. *ActiGraph White Paper: What is a Count?* URL: https://www.actigraphcorp.com/white-papers/. (accessed: 07.11.2019).

[3] Akin Avci et al. "Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey". In: *Proceedings of the 23th International conference on architecture of computing systems*. VDE. 2010, pp. 1–10.

[4] Ryan R Bailey, Joseph W Klaesner, and Catherine E Lang. "Quantifying real-world upper-limb activity in nondisabled adults and adults with chronic stroke". In: *Neurorehabilitation and neural repair* 29.10 (2015), pp. 969–978.

[5] João Carlos Alves Barata and Mahir Saleh Hussein. "The Moore–Penrose pseudoinverse: A tutorial review of the theory". In: *Brazilian Journal of Physics* 42.1-2 (2012), pp. 146–165.

[6] Elaine A Biddiss and Tom T Chau. "Upper limb prosthesis use and abandonment: a survey of the last 25 years". In: *Prosthetics and orthotics international* 31.3 (2007), pp. 236–257.

[7] Elaine Biddiss, Dorcas Beaton, and Tom Chau. "Consumer design priorities for upper limb prosthetics". In: *Disability and Rehabilitation: Assistive Technology* 2.6 (2007), pp. 346–357.

[8] Gabriele Bleser et al. "Cognitive learning, monitoring and assistance of industrial workflows using egocentric sensor networks". In: *PloS one* 10.6 (2015), e0127769.

[9] Gabriele Bleser, Gustaf Hendeby, and Markus Miezal. "Using egocentric vision to achieve robust inertial body tracking under magnetic disturbances". In: *In proceedings of the 10th IEEE International Symposium on Mixed and Augmented Reality*. IEEE. 2011, pp. 103–109.

[10] KC Border. "More than you wanted to know about quadratic forms". In: (2011).

[11] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. ISBN: 0521833787.

[12] Design Buzz. *19th century artificial limb from Victorian England looks menacingly creepy*. URL: https://designbuzz.com/19th-century-artificial-limb-from-victorian-england-looks-menacingly-creepy/. (accessed: 15.08.2020).

[13] John V Carlis and Joseph A Konstan. "Interactive visualization of serial periodic data". In: *Proceedings of the 11th annual ACM symposium on User interface software and technology*. 1998, pp. 29–38.

[14] Alix Chadwell et al. "The reality of myoelectric prostheses: understanding what makes these devices difficult for some users to control". In: *Frontiers in neurorobotics* 10 (2016), p. 7.

[15] Alix Chadwell et al. "Upper limb activity in myoelectric prosthesis users is biased towards the intact limb and appears unrelated to goal-directed task performance". In: *Scientific reports* 8.1 (2018), pp. 1–12.

[16] Alix Chadwell et al. "Visualisation of upper limb activity using spirals: A new approach to the assessment of daily prosthesis usage". In: *Prosthetics and orthotics international* 42.1 (2018), pp. 37–44.

[17] Henri De Vroey et al. "Hip and knee kinematics of the forward lunge one year after unicondylar and total knee arthroplasty". In: *Journal of Electromyography and Kinesiology* 48 (2019), pp. 24–30.

[18] Ethan Eade. *Lie groups for 2d and 3d transformations*. URL: http://www.ethaneade.org/lie.pdf. (accessed: 12.04.2020).

[19] Division of Engineering Brown University. *Lecture notes: Dynamics and Vibrations, 5 Dynamics of Rigid Bodies*. Oct. 2019.

[20] Jeroen D Hol. "Sensor fusion and calibration of inertial sensors, vision, ultra-wideband and GPS". PhD thesis. Linköping University Electronic Press, 2011.

[21] Berthold KP Horn. "Closed-form solution of absolute orientation using unit quaternions". In: *Josa a* 4.4 (1987), pp. 629–642.

[22] Manon Kok, Jeroen D Hol, and Thomas B Schön. "An optimization-based approach to human body motion capture using inertial sensors". In: *IFAC Proceedings Volumes* 47.3 (2014), pp. 79–85.

[23] Manon Kok, Jeroen D Hol, and Thomas B Schön. "Using inertial sensors for position and orientation estimation". In: *Foundations and Trends in Signal Processing* 11.1–2 (2017), pp. 1–153.

[24] Manon Kok and Thomas B Schön. "A Fast and Robust Algorithm for Orientation Estimation using Inertial Sensors". In: *IEEE Signal Processing Letters* 26.11 (2019), pp. 1673–1677.

[25] Alfred E Kritter. "Myoelectric prostheses." In: *JBJS* 67.4 (1985), pp. 654–657.

[26] Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006. ISBN: 0521862051.

[27] Alberto Leardini et al. "Human movement analysis using stereophotogrammetry: Part 3. Soft tissue artifact assessment and compensation". In: *Gait & posture* 21.2 (2005), pp. 212–225.

[28] J. K. Lee and T. H. Jeon. "IMU-Based but Magnetometer-Free Joint Angle Estimation of Constrained Links". In: *Proceedings of the IEEE SENSORS*. 2018, pp. 1–4.

[29] David Loudon and Malcolm H Granat. "Visualization of sedentary behavior using an event-based approach". In: *Measurement in Physical Education and Exercise Science* 19.3 (2015), pp. 148–157.

[30] Bartjan Maat et al. "Passive prosthetic hands and tools: A literature review". In: *Prosthetics and orthotics international* 42.1 (2018), pp. 66–74.

[31] Sebastian Madgwick. "An efficient orientation filter for inertial and inertial/magnetic sensor arrays". In: *Report x-io and University of Bristol (UK)* 25 (2010), pp. 113–118.

[32] Sebastian OH Madgwick, Andrew JL Harrison, and Ravi Vaidyanathan. "Estimation of IMU and MARG orientation using a gradient descent algorithm". In: *Proceedings of the IEEE international conference on rehabilitation robotics*. IEEE. 2011, pp. 1–7.

[33] MTw User Manual. "MTw Hardware". In: *MT Manager, Awinda Protocol* (2014).

[34] Markus Miezal, Bertram Taetz, and Gabriele Bleser. "On inertial body tracking in the presence of model calibration errors". In: *Sensors* 16.7 (2016), p. 1132.

[35] Markus Miezal et al. "A generic approach to inertial tracking of arbitrary kinematic chains". In: *Proceedings of the 8th international conference on body area networks*. 2013, pp. 189–192.

[36] Heather Montague. *Humans of TU Delft: Gerwin Smit*. URL: https://www.delta.tudelft.nl/article/humans-tu-delft-gerwin-smit. (accessed: 18.11.2019).

[37] Alvin L Muilenburg and Maurice A LeBlanc. "Body-powered upper-limb components". In: *Comprehensive management of the upper-limb amputee*. Springer, 1989, pp. 28–38.

[38] Engineering National Academies of Sciences, Medicine, et al. *The promise of assistive technology to enhance activity and work participation*. National Academies Press, 2017.

[39] Marika Noorkõiv, Helen Rodgers, and Christopher I Price. "Accelerometer measurement of upper extremity movement after stroke: a systematic review of clinical studies". In: *Journal of neuroengineering and rehabilitation* 11.1 (2014), p. 144.

[40] Fredrik Olsson and Kjartan Halvorsen. "Experimental evaluation of joint position estimation using inertial sensors". In: *Proceedings of the 20th International Conference on Information Fusion*. IEEE. 2017, pp. 1–8.

[41] Fredrik Olsson et al. "Joint axis estimation for fast and slow movements using weighted gyroscope and acceleration constraints". In: *Proceedings 22th International Conference on Information Fusion (FUSION)*. IEEE. 2019, pp. 1–8.

[42] Ottobock. *Body-powered prosthetic solutions*. URL: https://www.ottobockus.com/prosthetics/upper-limb-prosthetics/solution-overview/body-powered-prosthetic-solutions/. (accessed: 07.11.2019).

[43] Ottobock. *Michelangelo prosthetic hand*. URL: https://www.ottobockus.com/prosthetics/upper-limb-prosthetics/solution-overview/michelangelo-prosthetic-hand/. (accessed: 07.11.2019).

[44] Shyamal Patel et al. "A review of wearable sensors and systems with application in rehabilitation". In: *Journal of neuroengineering and rehabilitation* 9.1 (2012), p. 21.

[45] Monique Paulich et al. “Xsens MTw Awinda: Miniature wireless inertial-magnetic motion tracker for highly accurate 3D kinematic applications”. In: *Xsens: Enschede, The Netherlands* (2018).

[46] D Peach, Jacqueline Van Hoomissen, and H L Callender. “Exploring the ActiLife® filtration algorithm: Converting raw acceleration data to counts”. In: *Physiological measurement* 35.12 (2014), p. 2359.

[47] apc prosthetics. *Esthetique Silicone Program*. URL: http://apcprosthetics.com.au/high-definition-silicone-prostheses/. (accessed: 18.11.2019).

[48] Friedrich Pukelsheim. “The three sigma rule”. In: *The American Statistician* 48.2 (1994), pp. 88–91.

[49] John Ratcliffe. *Foundations of hyperbolic manifolds*. Vol. 149. Springer Science & Business Media, 2013.

[50] Angelo M Sabatini et al. “Assessment of walking features from foot inertial sensing”. In: *IEEE Transactions on biomedical engineering* 52.3 (2005), pp. 486–494.

[51] Jeffer E Sasaki, Dinesh John, and Patty S Freedson. “Validation and comparison of ActiGraph activity monitors”. In: *Journal of science and medicine in sport* 14.5 (2011), pp. 411–416.

[52] Thomas Seel, Thomas Schauer, and Jörg Raisch. “Joint axis and position estimation from inertial measurement data by exploiting kinematic constraints”. In: *Proceedings of the IEEE International Conference on Control Applications*. IEEE. 2012, pp. 45–49.

[53] Gerwin Smit, Dick H Plettenburg, and Frans CT van der Helm. “The lightweight Delft Cylinder Hand: first multi-articulating hand that meets the basic user requirements”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 23.3 (2014), pp. 431–440.

[54] Gerwin Smit et al. “A self-grasping hand prosthesis”. In: (2019).

[55] Sauer T. *Numerical Analysis*. Pearson, 2012.

[56] Stewart G Trost, Kerry L Mciver, and Russell R Pate. “Conducting accelerometer-based activity assessments in field-based research”. In: *Medicine & science in sports & exercise* 37.11 (2005), S531–S543.

[57] Catrine Tudor-Locke, William D Johnson, and Peter T Katzmarzyk. “Accelerometer-determined steps per day in US adults.” In: *Medicine and science in sports and exercise* 41.7 (2009), pp. 1384–1391.

[58] Roberto G Valenti, Ivan Dryanovski, and Jizhong Xiao. “Keeping a good attitude: A quaternion-based orientation filter for IMUs and MARGs”. In: *Sensors* 15.8 (2015), pp. 19302–19330.

[59] Michel Verhaegen and Vincent Verdult. *Filtering and system identification: a least squares approach*. Cambridge university press, 2007.

[60] Marc S Weinberg et al. “Tilt determination in MEMS inertial vestibular prosthesis”. In: *Journal of biomechanical engineering* 128.6 (2006), pp. 943–956.

[61] Ive Weygers et al. “Drift-Free Inertial Sensor-Based Joint Kinematics for Long-Term Arbitrary Movements”. In: *IEEE Sensors Journal* (2020).