NTNU

Norwegian University of
Science and Technology

Robotic rehabilitation of upper-limb after stroke

Implementation of rehabilitation control
strategy on robotic manipulator

Mads Johan Laastad

Master of Science in Cybernetics and Robotics
Submission date: June 2017
Supervisor: Øyvind Stavdahl, ITK
Co-supervisor: Erik Kyrkjebø, Høgskolen i sogn og fjordane

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Abstract

Globally, stroke is one of the main causes of permanent neurological damage [18]. Partial or total paralysis of the extremities is the most common complication, with paralysis in upper-limbs being the most prevalent. Efficient and available rehabilitation therapy is essential for the patient's recovery process. Traditional physical therapy is a resource intensive and commonly used approach. Research on robotic rehabilitation aims to provide a viable rehabilitation tool.

The main objective of this master thesis is to design and implement a safe real-time control system on an industrial six-axis manipulator with an external force/torque sensor for robotic upper-limb rehabilitation of stroke patients.

The chosen approach is to implement control strategies directly in the tool frame of the manipulator. This is achieved by utilizing the UR5 from Universal Robot and the Mini45 F/T sensor from ATI Industrial Automation. Two verification test are chosen based on activities of daily life (ADL). The best low-level control strategy is achieved by indirect force and torque control through a joint velocity interface.

The UR5 firmware operates with an unknown internal controller. An external controller is designed incrementally to investigate the unknown system dynamics and find the best possible low-level performance. Numerous safety mechanisms are added to the external controller. Four high-level control strategies are developed and implemented.

Three main safety-related challenges with robotic rehabilitation are identified. Two of them are related to and solved by the external force/torque sensor. The third challenge is related to the self-collisions inside the workspace of the UR5 manipulator. This challenge is also applicable to all six-axis robot manipulators. The three challenges are analyzed and solved with a safety-oriented approach.

The safety and functionality of the robotic rehabilitation system are experimentally verified. The behaviour of the rehabilitation modes is analyzed and discussed based on raw data and video recordings.

The conclusion is that robotic upper-limb rehabilitation of stroke patients utilizing the UR5 manipulator and the Mini45 F/T sensor is safe and feasible.

Sammendrag

Hjerneslag er en verdensledende årsak til permanente nevrologiske skader [18]. Delvis eller total lammelse i ekstremiteter er den vanligste komplikasjonen etter et hjerneslag, hvor lammelser i armene er mest utbredt. Effektiv og tilgjengelig rehabiliteringsterapi er essensielt for å gjenopprette funksjonaliteten. Tradisjonell fysioterapi er en ressurskrevende og utbredt metode. Forskning innen robot rehabilitering har som mål å tilby et levedyktig rehabiliterings verktøy.

Hovedformålet med denne masteroppgaven er å designe og implementere et trygt, sanntidssystem for kontroll av en industriell seks-akset manipulator med en ekstern kraft/-dreiemoment sensor til bruk i robot rehabilitering av armene til slagpasienter.

Den foretrukne metoden involverer å implementere kontroll strategiene direkte i verktøyets referansesystem. Dette blir realisert vha. en UR5 fra Universal Robot og en Mini45 F/T sensor fra ATI Industrial Automation. To verifiserings tester blir valgt basert på dagligdagse aktiviteter. Den beste oppnåelige lav-nivå kontrollen blir realisert vha. et ledd-hastighet grensesnitt.

Programvaren til UR5 inneholder en ukjent indre kontroller. En ytre kontroller blir utviklet stegvis for å undersøke den ukjente dynamikken og for å finne den beste lav-nivå ytelsen. Flere sikkerhetsmekanismer blir implementert i den ytre kontrolleren. Fire høy-nivå kontroll strategier blir utviklet og implementert.

Tre sikkerhetsrelaterte hovedutfordringer ved robot rehabilitering blir identifisert. To av dem er relatert til og blir løst av den eksterne kraft/dreiemoment sensoren. Den tredje hovedutfordringen er relatert til selv-kollisjoner i arbeidsområdet til UR5 manipulatoren. Denne utfordringen er relevant for alle seks-akse robot manipulatorer. De tre utfordringene blir analysert og løst med en sikkerhetsorientert tilnærming.

Sikkerheten og funksjonaliteten til robot rehabiliterings systemet blir verifisert gjennom eksperimenter. Oppførselen til rehabiliterings modusene blir analysert og diskutert basert på innsamling av rådata og videoopptak.

Konklusjonen er at robot rehabilitering av armene til slagpasienter vha. en UR5 robot manipulator og en Mini45 F/T sensor er trygt og gjennomførbart.

PROJECT ASSIGNEMENT

Robotic rehabilitation of upper-limb after stroke

BACKGROUND

Approximately 15 000 people in Norway experience a stroke each year, and the incidence is believed to increase by up to 50 per cent in the next 20 years due to an ageing population. Stroke patients require extensive rehabilitation training to recover. Intense repetitions of coordinated motor activities in rehabilitation therapy and training constitute a significant burden for therapists, and due to economic reasons the duration of primary rehabilitation is getting shorter and shorter.

Recent systematic reviews of trials comparing conventional rehabilitation therapy by physiotherapists (CT) with robotassisted therapy (RT), suggests that RT gives similar upper-limb rehabilitation in terms of motor function recovery, activities of daily living (ADL) and motor control. Also, extra sessions of RT in addition to regular CT have been shown to be more beneficial than regular CT alone in motor recovery of stroke patients [1]. Robot rehabilitation of upper-limb after stroke have thus gained increasing interest in recent years, and new robotic devices have been designed to assist in physical rehabilitation of stroke patients [2].

These robot rehabilitation devices may significantly improve on existing practice through increased effectivity in rehabilitation training with less man-power, closer monitoring of patient progress, higher quality and more objective therapeutic sessions, and more personalized follow-up for each individual patient.

TASK

A number of different control strategies have been employed for robot rehabilitation ranging from high-level strategies – such as assistive, challenge-based, haptic or couching strategies designed to provoke motor plasticity and thus improve motor recovery – to low-level strategies to control force, position, impedance and admittance factors of the high-level strategies [2].

Recent advancements in light-weight and affordable robot manipulators that complies with safety regulations can operate without safety guards between robots and humans, and have a potential for decreasing the cost of custom built robotic rehabilitation devices significantly. The project shall propose a robot control strategy for rehabilitation of upper-limb after stroke using a standard industrial robot manipulator. The proposed robot manipulator is the UR5 robot from Universal Robots. The proposed strategy should be verified in simulation and experiments.

Task description

1. Choose a suitable external force/torque sensor and install it on the UR5 robot manipulator from Universal Robots
2. Choose and implement a robotic control interface that enables low-level control strategies to control force, position, impedance and admittance factors
3. Choose and implement a low-level control strategy for robotic rehabilitation on the UR5 robot manipulator from Universal Robots
4. Analyze the low-level control strategy performance though experimental verification
5. If time permits: Implement high-level control strategies for robotic rehabilitation, analyze the high-level robot rehabilitation control strategies in experiments.

Useful resources may be the Matlab Robotics System Toolbox and the V-REP robot simulator.

Objective and purpose
The objective of the master thesis is to design and implement a chosen robot rehabilitation control system on the UR5 robot manipulator from Universal Robots for upper-limb stroke patients.

Subtasks and research questions

- Is implementation of a robotic rehabilitation system with the Universal Robots UR5 robot manipulator feasible?
- What are the main challenges in robotic rehabilitation using Universal Robots UR5 robot manipulator with an external force/torque sensor?

Main supervisor: Associate professor Øyvind Stavdahl MSc, Ph.D., Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU).

Co-supervisor: Vice-rector Erik Kyrkjebø MSc, Ph.D., Western Norway University of Applied Sciences.

Bibliography

- *Norouzi-Gheidari, N., P.S. Archambault, and J. Fung, Effects of robot-assisted therapy on stroke rehabilitation in upper limbs: Systematic review and meta-analysis of the literature. J. of Rehab. R&D, 2012. 49(4): p. 479-495.*
- *Maciejasz, P., et al., A survey on robotic devices for upper limb rehabilitation. J Neuroeng Rehab, 2014. 11(1).*

Preface

This master thesis was written during the spring of 2017 as part of the course *"TTK4900 Teknisk kybernetikk, masteroppgave"* at the Norwegian University of Science and Technology (NTNU). This course corresponds to a workload of 30 ECTS credits, equivalent to an entire semester workload.

Thank you to Erik Kyrkjebø MSc, Ph.D. and Øyvind Stavdahl MSc, Ph.D. for supervising this master thesis. Thank you to SINTEF for providing a suitable force/torque sensor. Thank you to student colleagues, Ph.D. candidates and professors at the Department of Engineering Cybernetics at NTNU for all the help, advice and encouragement. Thank you to Tom Meland Pedersen, Christina Laastad and Bjørn Johan Laastad for proofreading the master thesis, your work is highly appreciated.

Contents

Abstract i

Sammendrag ii

Assignment iii

Preface v

Table of Contents viii

Abbreviations ix

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Thesis goals . . . 2
- 1.3 Attitude and approach . . . 2
- 1.4 Contributions . . . 3
- 1.5 Thesis outline . . . 4

2 Background 6
- 2.1 Stroke rehabilitation strategies . . . 6
  - 2.1.1 Assistive vs. passive therapy . . . 6
  - 2.1.2 Constraint-induced movement therapy . . . 6
  - 2.1.3 Exclusion criteria . . . 7
- 2.2 Experimental verification . . . 7
  - 2.2.1 Functional reach test . . . 7
  - 2.2.2 Drinking test . . . 8
  - 2.2.3 Video recording . . . 8
- 2.3 Robotic manipulator . . . 8
- 2.4 External force/torque sensor . . . 8
- 2.5 Summary . . . 9

3 System overview **10**
3.1 Universal Robots UR5 . . . 10
3.1.1 Control methods . . . 11
3.2 ATI Mini45 force/torque sensor . . . 13
3.3 End-effector design . . . 14
3.4 External computer . . . 18
3.5 Stationary structural hardware . . . 19
3.6 Complete system setup . . . 20
3.6.1 Laboratory safety . . . 21
3.6.2 End-effector design . . . 22
3.6.3 System delay . . . 23
3.7 Summary . . . 25

4 Manipulator modeling **26**
4.1 Kinematics . . . 26
4.1.1 Forward kinematics . . . 26
4.1.2 Inverse kinematics . . . 28
4.2 Velocity kinematics . . . 29
4.2.1 Geometric Jacobian . . . 29
4.2.2 Manipulator Jacobian . . . 29
4.2.3 Inverse velocity kinematics . . . 30
4.2.4 Force control . . . 30
4.3 Summary . . . 31

5 Stroke patients **32**
5.1 Sampling frequency . . . 32
5.2 Implementation challenges . . . 34
5.2.1 Unwanted oscillations . . . 34
5.2.2 Mounting bias . . . 36
5.2.3 Workspace limitations . . . 38
5.3 Executive controller security . . . 40
5.3.1 Internal controller safety mechanisms . . . 40
5.3.2 External controller security mechanisms . . . 41
5.4 Summary . . . 43

6 Controller design **44**
6.1 Gravity compensation . . . 44
6.2 Executive controller design . . . 46
6.3 Low-level control strategy . . . 47
6.3.1 Force control . . . 51
6.3.2 Torque control . . . 54
6.3.3 Summary . . . 58
6.4 High-level control strategies . . . 59

7 Experimental verification **63**
7.1 Emergency shutdown procedures . . . 63
7.1.1 Virtual workspace . . . 64
7.2 Robotic rehabilitation modes . . . 70
7.2.1 Data and video recording . . . 70
7.2.2 2-plane mode . . . 70
7.2.3 Buoyancy mode . . . 73
7.2.4 Resistance mode . . . 77
7.2.5 Random mode . . . 80
7.3 Summary . . . 83

8 Discussion **85**
8.1 Patient security . . . 85
8.2 Robotic rehabilitation modes . . . 86
8.2.1 Compliance mode . . . 86
8.2.2 2-plane mode . . . 86
8.2.3 Buoyancy mode . . . 87
8.2.4 Resistance mode . . . 87
8.2.5 Random mode . . . 88

9 Conclusion and Future work **89**
9.1 Conclusion . . . 89
9.2 Future work . . . 91
9.2.1 End-effector design . . . 91
9.2.2 Control signal filtration . . . 91
9.2.3 GUI . . . 91
9.2.4 Rehabilitation exercises . . . 92
9.2.5 Expanding the system . . . 92

Bibliography **92**

Appendix A Source code **96**

Appendix B Miscellaneous **99**
B.1 Universal Robot UR5 . . . 99
B.1.1 Denavit-Hartenberg convention . . . 99
B.1.2 Reference frames for UR5 robot manipulator . . . 100
B.1.3 URScript motion commands . . . 101
B.1.4 Configuration of the UR5 robot manipulator . . . 102
B.2 Low-pass filtration in the Mini45 F/T sensor . . . 102
B.3 Additional experimental verification tests . . . 104

Appendix C Safety documentation **105**

Abbreviations

6DOF = Six degrees of freedom

ADL = Activities of daily life

API = Application programming interface

CB = Control box

OS = Operating system

GUI = Graphical user interface

ICMP/IP = Internet control message protocol/Internet protocol

RTDE = Real-time data exchange

TCP = Tool center point

TCP/IP = Transmission control protocol/Internet protocol

UDP/IP = User datagram protocol/Internet protocol

UR3 = Universal robot model 3

UR5 = Universal robot model 5

UR10 = Universal robot model 10

Chapter 1

Introduction

1.1 Motivation

Stroke is the leading cause of permanent neurological damage and one of the leading causes of death among adults [11] in Norway with over 15 000 confirmed incidents annually. On a global scale, about 15 million people suffer from stroke annually, with over 5 million casualties and another 5 million left with permanent neurological damage [18]. The Norwegian Directorate of Health [11] estimate that $60-80\%$ of the stroke patients will experience complications in the immediate aftermath of a stroke. That number increases to above 90% after $3-6$ months. Common complications include; paralysis, epileptic seizure, falling, spasticity, swallowing difficulties and shoulder pain. These complications often result in reduced muscle strength, reduced motion control, reduced movement speed, disturbed coordination and increased rate of muscle fatigue. Paralysis is one of the most serious consequences after a stroke, having a crippling affect on the patient's daily life. Partial or total paralysis of the extremities occurs in more than 80% of the stroke patients [11]. The severeness and prevalence of paralysis make it one of the biggest challenges related to stroke rehabilitation and is the focus of a robotic rehabilitation project pioneered by Erik Kyrkjebø, Vice-rector of R&D, Sogn og Fjordane, Western Norway University of Applied Sciences, Norway.

Traditional rehabilitation of the upper-limbs after a stroke is usually provided by physical therapists. This treatment has proven to be effective, but it is resource intensive and cannot keep up with the growing demands. Robotic rehabilitation can hopefully become a viable, affordable and safe rehabilitation alternative. Several studies [19] [15] have already shown that robotic rehabilitation, utilizing customized and expensive robotic exoskeletons, can rival traditional rehabilitation in terms of quality. The robotic rehabilitation system hopes to illustrate how an inexpensive industrial robot can be used safely in the rehabilitation of stroke patients as a valuable rehabilitation tool in the traditional treatments. If successful, the system will reduce the costs associated with stroke rehabilitation, and more importantly improve the stroke patient's quality of life.

Figure 1.1: Example of robotic rehabilitation with stroke patients using an exoskeleton device called ArmeoPower. Figure taken from [16].

1.2 Thesis goals

The main objective of this thesis is to design and implement a safe real-time robot rehabilitation control system on the UR5 robot manipulator from Universal Robots with an external force/torque sensor for rehabilitation of upper-limbs in stroke patients.

The secondary objectives of the thesis includes determining the feasibility of the robotic rehabilitation system with the utilization of the UR5 robot manipulator and an external force/torque sensor. Additionally, the thesis aims to identify and investigate the main implementation challenges related to utilizing a standard robotic manipulator with an external force/torque sensors in the robotic rehabilitation system.

1.3 Attitude and approach

The thesis assignment provides several concrete steps in the task description towards reaching the goals and objectives. These steps are fairly self-explanatory and leave little room for interpretation. However, the large scope of the thesis goals and objectives introduces some leeway on determining the preferred approach and general attitude towards the design and implementation of solutions to the different challenges.

Human-machine interaction

The assignment and thesis goals desire a close interaction between an industrial robot manipulator and stroke patients. This delicate human-machine interaction requires a security-oriented approach to the human-machine interaction. Therefore, an initial design decision is to implement the control feedback loop directly in the tool frame, defined by the current

orientation and position of the manipulator's end-effector. This interaction facilitates for a natural force and torque transfer between the stroke patients and the manipulator.

Practicality and modularity

The interpretation of a probable progress scenario for the robotic rehabilitation system is through an incremental implementation and introduction into the Norwegian healthcare system. Firstly, the system needs to be thoroughly tested and experimentally verified in a laboratory setup. Then other engineers, both future master students and external experts, needs to independently verify the safety of the system. The clinical ethics committee at St. Olavs University Hospital must review the independent verification before the system can be approved for human trials. If the system is approved it can be enrolled into scientific studies with small control groups consisting of actual stroke patients. If the results are promising the system can be subjected to a long-term study at the St. Olavs University Hospital. Additional rounds of incremental improvements and increased security hopefully result in a system that can be utilized by the stroke patients without direct supervision. The ultimate goal is a completely mobile system with remote supervision capabilities that can be borrowed upon request by individual stroke patients and utilized in the patient's own home.

The system is currently in the first stage of development within this progress scenario. The systems intended use as a rehabilitation tool implies referring to the physical therapists as *"users"* and referring to stroke patients directly or sometimes as *"end-users"*. The author of this master thesis is referred to as *"operator"* during testing and development. The guiding attitude in the design process is mainly focused on providing safe functionality as the foundations for the robotic rehabilitation system is established. The current status also encourages a modular and dynamic approach towards realizing the thesis goals and objectives. This is seen as a significant advantage when several external parties seek to examine or replicate the results generated by this master thesis. Some implementations are affected by the safety oriented attitude as simplifications and a highly practical approach is necessary to ensure a safe and steady progress in the development. Whenever relevant, implementation challenges that require simplifications is presented together with an analysis and discussion of the situation.

1.4 Contributions

This thesis presents the design and implementation of a safe and complete real-time robotic rehabilitation system utilizing a standard robotic manipulator and an external force/torque sensor. Several contributions are directly related to the task description presented in the master thesis assignment.

1. An analysis of the preconditions and available hardware result in choosing the Mini45 force/torque sensor from ATI Industrial Automation as a suitable external force/-torque sensor. The sensor is installed on the UR5 robot together with several end-effector prototypes.

2. An analysis of the UR5 robotic control interface determine that the newly developed ur_modern_driver based on URScript commands exposes a joint velocity interface that provides the best possible robotic control interface. The control interface enables the implementation of a low-level control strategy in the form of indirect force- and torque control in the end-effector tool space.

3. An analysis of the joint velocity control interface determine that streaming URScript speedj commands through an external controller provide the best method for implementing indirect force- and torque control in the end-effector tool space.

4. A thorough and incremental low-level controller design process provide the best possible solution for implementing the external controller. The low-level control strategy is implemented as a split P-controller design with two sets of experimentally tuned gain parameters and is named compliance mode.

5. Four high-level control strategies for robotic rehabilitation is based on the low-level control strategy. Manipulating the control inputs results in four rehabilitation modes with different behaviour aimed at stroke patients at different stages of the rehabilitation progress. The four rehabilitation modes are named 2-plane mode, buoyancy mode, resistance mode and random mode. Experimental verification of the four rehabilitation modes is presented together with supportive video recordings.

Other contributions include designing and implementing several end-effector prototypes. Performing a risk assessment of the intended application and interaction between the UR5 robot manipulator and humans. Determining several parameters related to stroke patient including exclusion criteria, sampling frequency and control signal filtration. Estimating the total system delay based on the laboratory setup. Designing, implementing and experimentally verifying external security mechanisms including several emergency shutdown procedures and a virtual workspace implemented with soft boundaries.

The design and implementation of the robotic rehabilitation system identifies, analyzes and solves numerous implementation challenges. Some smaller implementation challenges that are not deemed to have a significant and negative impact on the system security is presented in the final chapter as suggestions for future work. The work presented in this thesis ultimately concludes that the robotic rehabilitation system developed with the Universal Robot UR5 and the Mini45 F/T sensor is both feasible and safe.

1.5 Thesis outline

The thesis structure seeks to follow the introduction, methods, results, and discussion(IMRAD) structure [27]. However, in some design decisions it natural to draw smaller conclusions based on the preliminary results. This leads to some slight deviations from the established IMRAD structure and an outline of the thesis structure is therefore presented.

Chapter 2 outlines an analysis of relevant stroke rehabilitation strategies. Two experimental verification tests are suggested. The precondition behind utilizing the UR5 robot manipulator is discussed and a suitable force/torque sensor is chosen.

Chapter 3 describes and analyze the utilized hardware and proprietary software. Several end-effector prototypes are developed and analyzed and an overview of the robotic rehabilitation system architecture is presented. The total system delay in the laboratory setup is estimated.

Chapter 4 presents the mathematical modeling of the UR5 robotic manipulator. Forward- and inverse kinematic solutions together with the Jacobian is presented and discussed.

Chapter 5 presents an analysis of the sampling frequency, signal filtration and end-effector restrictions required for working with stroke patients. The security mechanisms provided by the internal controller is presented and newly developed security mechanisms provided by the external controller is designed and discussed.

Chapter 6 presents the development, analysis and performance of the low-level control strategy provided by the external controller. The design of a low-level control strategy is experimentally tuned and thoroughly tested. The four high-level control strategies are presented.

Chapter 7 conducts experimental verification testing of the external security mechanisms and the four high-level control strategies. The results are analyzed and discussed.

Chapter 8 presents a general discussion that is naturally oriented around the safety of the robotic rehabilitation system and the four high-level control strategies.

Chapter 9 concludes the thesis by summarizing the obtained results and answering the research questions posed by the thesis assignment. Suggested topics for future development of the robotic rehabilitation system is presented.

Appendix A presents the file structure and file description for the source code utilized to create the robotic rehabilitation system. The steps needed to run the source code is presented in a brief start-up manual.

Appendix B briefly presents an experimental verification test not included in the main text. Technical details about the UR5 and URScript commands is presented together with the lox-pass filter attenuation and filtration delay for the ATI Mini45 F/T sensor.

Appendix C describes and analyze a risk assessment for the application of the UR5 in this master thesis. The TÜV safety classification for Universal Robots UR5 manipulator is presented.

Chapter 2

Background

This chapter presents and analyzes relevant stroke rehabilitation strategies. General guidelines for the design and development of the robotic rehabilitation system is compiled through analysis of preliminary work and personal conversations with a rehabilitation expert. Experimental verification tests, preliminary system requirements, available robotic manipulator hardware and sensory hardware is discussed and analyzed.

2.1 Stroke rehabilitation strategies

2.1.1 Assistive vs. passive therapy

There are two main approaches towards implementing robotic rehabilitation on upper-limbs in stroke patients, they are known as active- and passive therapy. As the name suggests, passive therapy does not require the patient to take an active part in the movement as the patient is passively guided through a specific movement by the robot. Active therapy represents the opposite approach as it requires constant participation of the patient to maintain a movement or perform a task. The approach utilized in this master thesis is active therapy as it provides the most potential for a robotic rehabilitation system [7] [6].

2.1.2 Constraint-induced movement therapy

”Constraint-induced movement therapy” is a common rehabilitation strategy in physical therapy that involves reducing the degree of freedom in the afflicted limb in order to retain some degree of functionality. The strategy is based on the human body’s natural reaction to trauma or injuries. For example, a patient suffering from trauma or injuries in the lower limbs will naturally adopt a compensating gait pattern with fewer degrees of freedom in order to retain locomotive mobility. This is well documented in the gait pattern of elderly suffering from hip fractures [20] [17]. The limited functionality enabled by the constraint induced movement therapy then becomes a platform for rebuilding muscle strength, a more

complex repertoire of functional movements and ultimately expand the rehabilitation efforts into other rehabilitation strategies.

The robotic rehabilitation system was discussed with Torunn Askim Ph.D., associate professor at Department of Neuroscience at the Faculty of Medicine NTNU, during a lecture [8] and subsequent meeting in October 2016. The recommended approach is to not expose stroke patients to constraint-induced movement therapy. The recommendation is based on a lack of scientific documentation on the positive effect constraint-induced movement therapy have on stroke patients.

2.1.3 Exclusion criteria

Professor Torunn Askim Ph.D. also recommended limiting the stroke patients eligible for robot rehabilitation treatment. This would avoid rehabilitation complications caused by weak stroke patients in early rehabilitation stages with severe symptoms and minimal functionality. The suggested exclusion criteria is a required minimal wrist functionality of $\pm 10^{\circ}$ rotation in all three rotational degrees of freedom. Stroke patients unable to perform these rotational movements should be excluded from robot rehabilitation treatment until traditional stroke rehabilitation enable them to display the required minimal wrist functionality.

2.2 Experimental verification

Traditional rehabilitation methods focus on regaining functionality related to activities of daily life (ADL). Normally, ADL is separated into basic ADL and instrumental ADL. Basic ADL consists of self-care tasks while instrumental ADL focuses on the individual patient's ability to function independently in a community. Examples of basic ADL's include bathing, dressing, functional mobility, personal hygiene and eating without assistance. Examples of instrumental ADL's include managing money, cleaning the house, preparing meals, using communication tools i.e. telephone, computers, etc. and visiting a doctor. It is evident that most instrumental ADL's is outside the scope of the robotic rehabilitation system, the focus is therefore on basic ADL's.

2.2.1 Functional reach test

The approach in traditional stroke rehabilitation is to break down complicated ADL's into a set of simpler tasks or movements that together constitute the entire ADL task or movement. The most important subset functionality to regain after a stroke is a *"functional reach"*. Generally, a functional reach can be described as an abduction- and adduction movement by the upper-limbs in the sagittal-, frontal- or transverse plane. The functional reach is the foundation for almost all ADL and is therefore chosen as an experimental verification test for the robotic rehabilitation system.

2.2.2 Drinking test

Eating without assistance is a basic ADL that greatly improve the independence and quality of life for stroke patients. Regaining the ability to eat without assistance if often an early goal in traditional stroke rehabilitation. Drinking from a cup is a subset of eating without assistance it is therefore chosen as an experimental verification test for the robotic rehabilitation system.

2.2.3 Video recording

The robotic rehabilitation system seeks to encourage a natural interaction between an industrial robot and stroke patients. Analyzing raw data from the two suggested experimental verification tests provide a crucial scientific part of the interaction analysis. Capturing the experimental verification on video could provide another analytic dimension and possibly a deeper understanding of the delicate interaction. The two suggested experimental verification tests should seek to capture the results on video and present this together with the data analysis.

2.3 Robotic manipulator

The robotic rehabilitation system is intended to be based around an industrial 6-DOF robot manipulator. Choosing a suitable robotic manipulator is a decisive decision that affects the implementation and feasibility of the robotic rehabilitation system. The delicate interaction between an industrial robot and stroke patients severely limit the number of applicable robotic manipulators. Potential candidates need to pass several safety classifications in order to operate in close approximation to humans without the requirements of a safety cage.

The UR5 robot manipulator from Universal Robot is one of the few industrial robot manipulators that provide the required safety classifications. The decision to develop the robotic rehabilitation system on the UR5 robot manipulator is a precondition in this master thesis. The precondition is the result of an intersection between the limited number of applicable and accessible robotic manipulators. Determining the feasibility of the robotic rehabilitation system implemented on the UR5 robotic manipulator is an important research question in this master thesis.

2.4 External force/torque sensor

The robotic rehabilitation system seeks to emphasize and encourage the natural and delicate interaction between an industrial robot and stroke patients. The sensor utilized for providing a control feedback loop between the manipulator and stroke patients is a crucial part of this interaction. Implementing the control feedback loop directly in tool space, defined by the current orientation and position of the manipulator's end-effector, is a design choice that facilitates a natural interaction with the stroke patients.

The safety functionality in the UR5 relies on the ability to detect interactions between

the robot manipulator and the environment. The detection is captured by servo based internal sensors. The internal sensors is able to detect crude safety related interactions in a reliable manner, but they are not suited to detect the light and delicate interactions between the manipulator and stroke patients. The servo-based internal sensors are unsuited for a control feedback loop in tool space due to sensory blind-spots, inaccuracy, lack of documentation and unsupported direct access to the raw sensor data. It is clear that an external force/torque sensor is required.

The number of available external force/torque sensors owned by the Department of Engineering Cybernetics at NTNU is limited. Luckily, an external sensor is made available through a collaboration between the cybernetics department and the independent research organization SINTEF. The external sensor is the Mini45 force/torque sensor from ATI Industrial Automation. The high force and torque data resolution provided by the Mini45 F/T sensor is capable of capturing the light and delicate interaction between the manipulator and stroke patients. The Mini45-E transducer is small and capable of being mounted on the end-effector to facilitate the desired tool space oriented feedback control loop. The Mini45 F/T sensor from ATI is deemed suitable as external sensor utilized in this master thesis. Determining the main challenge of the robotic rehabilitation system implemented with an external force/torque sensor is an important research question in this master thesis.

2.5 Summary

This chapter has presented and analyzed relevant stroke rehabilitation strategies. The most relevant rehabilitation strategy for robotic rehabilitation is based on assistive therapy. The recommendation that caution towards implementing constraint-induced movement therapy is presented together with a suggested exclusion criteria for stroke patients applicable for robotic rehabilitation treatment.

Two experimental verification tests, a functional reach- and drinking test, is presented. The verification tests should seek to improve an analysis understanding of the human-robot interaction by complementing the raw data with video recordings.

The utilization of the UR5 robot manipulator from Universal Robot has been presented together with the main reasoning behind the precondition.

The internal servo based sensors in the UR5 is deemed unsuited for a tool space oriented feedback control loop. The Mini45 force/torque sensor from ATI Industrial Automation is presented as a suitable external sensor with the desired performance and ability to realize the desired tool space feedback loop.

Chapter 3

System overview

This chapter establishes the hardware and software foundation for the robotic rehabilitation system. The chosen robotic manipulator and sensory hardware discussed in the previous chapter is presented in greater detail. The best method for controlling the robot manipulator is presented and discussed. The design of several end-effector prototypes is presented, analyzed and discussed. The mounting pedestal and associated table is presented. The complete hardware and software architecture is described in detail and implemented in a laboratory setup. Laboratory safety for testing and development is analyzed and discussed. The estimate for the total system delay is presented and analyzed.

3.1 Universal Robots UR5

The Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU) provided a UR5 robot for use in this master thesis. The robot is running a 2^{nd} generation internal controller called CB2, with software version 1.6.90722. The UR5 robot from Universal Robots is a relatively inexpensive 6-axis industrial manipulator with a reach of 850 mm and a maximal payload of 5 kg. The robot manipulator, Controller Box (CB) and teach pendant make up the entire robotic system illustrated in figure 3.1. Further details about the six rotational joints that together form the robot manipulator is illustrated in Appendix figure B.3.

Separating the UR5 robot from most six-axis industrial manipulators is the fact that it complies with point 5.10.5 of the EN ISO 10218-1:2006 standard, implying that the UR5 robot can operate in operation in close proximity to humans without a safety cage. This is possible due to a number of built-in safety mechanisms, most importantly a protective stop that triggers if a joint experience an external force that exceeds $150N$. More details about the limiting safety functions is described in table 5.2.

Figure 3.1: System overview of the UR5 robot from Universal Robots. The system consists of three main components. The six-axis robot manipulator on the left, the Controller Box centered on the table and the teach pendant attached to the front of the Controller Box. Figure taken from [30].

3.1.1 Control methods

There are three different ways of programming the UR manipulators; through the teach pendant using touch screen GUI called Polyscope, using scripts generated with URScript or through a C-API. Common to all three control methods is a maximum joint servo updating rate that is hardware limited to $125Hz$. The control hierarchy is illustrated in figure 3.2.

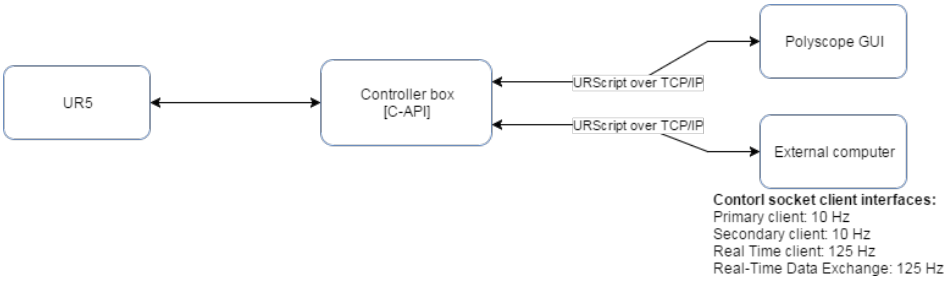


Figure 3.2: Control hierarchy of the UR5.

3.1.1.1 C programming

Universal Robot provides a C library for direct control of the speed and position of all joint servos in the robot. Though a direct inquiry to Universal Robots it is possible to get

insight into the C-API documentation. The documentation reveals that the C-API does not support direct low-level joint torque control. The limited control options is probably a security measure to prevent an accidental override of the internal firmware safety mechanisms described in table 5.2. Implementations of a custom external controller will be limited to an updating rate of 125 Hz.

The C-API library is not maintained, poorly documented and future software versions of the internal controller will not support this library. There is also a lack of properly maintained supportive code based on the C-API. The additional workload required for updating an existing driver or writing a new one from scratch is substantial. Together with the lack of direct joint torque control any potential advantages from using this library is eliminated. An overall assessment is that the C-API is not suitable for use in this master thesis.

3.1.1.2 Teach pendant

The teach pendant uses a touch screen graphical user interface (GUI) called Polyscope. It enables the user to program the robot sequentially through a combination of URScript commands and moving (”teaching”) the robot manipulator different waypoints. Polyscope can be a powerful tool for fast implementation, debug and prototyping. It is mainly developed for users without a programming background. The lack of basic functionality provided by the Polyscope interface makes the teach pendant unsuitable as a programming basis for this master thesis.

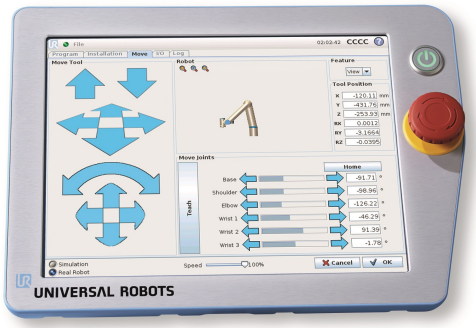


Figure 3.3: Universal Robot teach pendant. Figure taken from [21].

3.1.1.3 URScript

URScript is a robot programming language developed by Universal Robot. As it was developed specifically for controlling their robot manipulators it is constantly being maintained and developed. It is possible to interact and control the robot manipulator by establishing a TCP/IP socket connection between an external computer and the internal controller. The different runtime frequencies of the control interfaces are listed in figure 3.2, and is limited to a maximum updating rate of $125Hz$. Several drivers written in modern programming languages, like Python and C++, are based on this socket connection.

A review of commonly used drivers based on URScript is presented in [14]. The review concludes that the *ur_modern_driver*[2][3] developed by Thomas Timm Andersen, a robotics engineer from the Technical University of Denmark as part of his Ph.D. thesis on sensor-based real-time control of robots, is the candidate best suited for the intended application. Crucial arguments in this assessment include the driver’s ability to utilize the

maximum update rate of $125Hz$, backward comparability, automatic firmware detection and the fact that it currently is the most recently developed driver for the UR manipulators. Another desired feature is several safety features intended to prevent user generated errors from being executed.

The URScript command speedj enables the user to directly set the manipulator joint speeds. Constantly streaming this command though the ur_mordern_driver ensures the best possible performance from the UR5 robot and is therefore chosen as the basis for implementing indirect force and torque control.

3.2 ATI Mini45 force/torque sensor

The Mini45 force/torque sensor from ATI Industrial Automation is made available through a collaboration between the department of Engineering Cybernetics at NTNU and the independent research organization SINTEF. The force/torque sensor consists of F/T Net Box that process force and torque readings from a Mini45-E transducer shown in figure 3.4. The F/T Net Box draws power and communicates though a power over Ethernet (PoE) switch. This enables users to establish a socket connection between the F/T sensor and an external computer.

Figure 3.4: Mini45-E transducer. Picture taken from [5].

There is also a network configuration interface that enables the user to specify various sensor settings e.g. broadcasting frequency, calibration settings, bias vectors, unit prefix, etc. The available broadcasting frequency ranges from $1Hz$ to $7000Hz$. Low pass signal filtration with several predetermined cut-off frequencies ranging from $5Hz$ to $838Hz$ is available. The F/T sensor is setup with a metric calibration were the sensing ranges is well above the UR5 robot's maximum force load of 150N. The calibration details can be seen in table 3.1.

Calibration	**Sensing ranges**				**Resolution**			
	F_x, F_y	F_z	T_x, T_y	T_z	F_x, F_y	F_z	T_x, T_y	T_z
SI-290-10	290 N	580 N	10 Nm	10 Nm	$\frac{1}{4}$ N	$\frac{1}{4}$ N	$\frac{1}{188}$ Nm	$\frac{1}{376}$ Nm

Table 3.1: Metric calibration details of the ATI Mini45 force/torque sensor [4].

3.3 End-effector design

The physical interaction between stroke patients and the industrial robot is of crucial importance in this master thesis. Therefore, several end-effector prototypes that connect the robot and human through the force/torque sensor is developed.

3.3.0.1 Prototype 0.1

During early development, a simple end-effector with a high level of security was preferred. The resulting end-effector design was a plastic rod connected to the force/torque sensor with a fitting disk crafted from an aluminum block. The design enabled the operator to simply grasp the plastic rod and naturally exert forces and torques upon the robot through the sensor. If any dangerous situations occurred during early testing and development of the external controller the operator could quickly and safely release the grasp and detach from the robot. Dimensions and details are illustrated in figure 3.5.

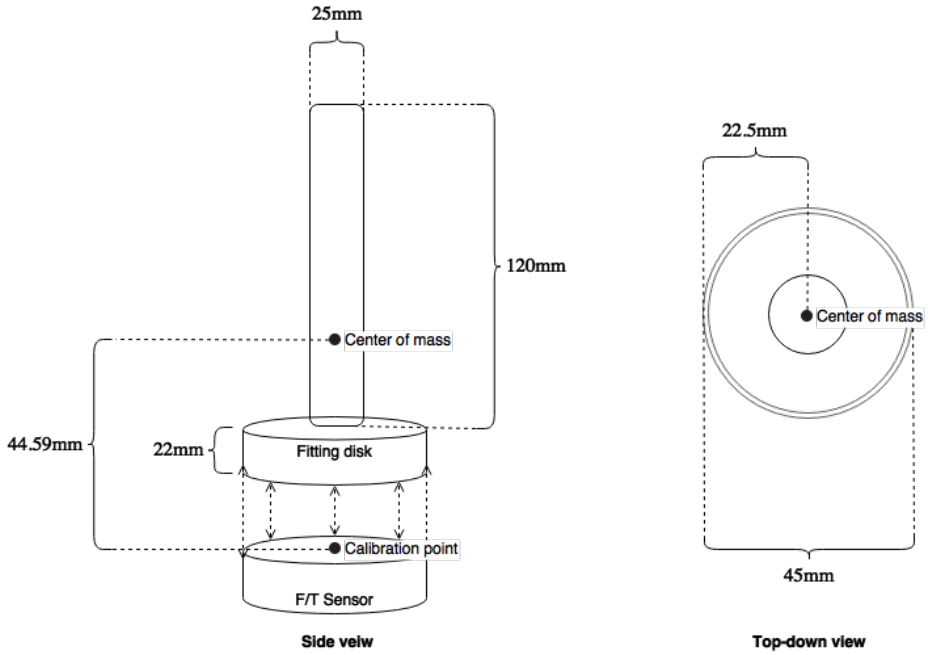


Figure 3.5: Prototype 0.1 of hand splint interface with the center of gravity highlighted.

Center of mass for Prototype 0.1

A good estimate of the center of mass ensured more accurate force/torque readings with less impact from the weight of the equipment upon the data readings. Some assumptions have been made in order to calculate the center of mass. Firstly, the weight of the aluminum fitting disk and plastic rod is assumed to be uniformly distributed. Secondly,

gravity is assumed to be homogeneously uniform for the entire system. Thirdly, the aluminum fitting disk and the plastic rod is assumed to be two perfect cylinders with radius $22.5mm$, $12.5mm$ and height $22mm$, $120mm$ respectively. With these assumptions, we can simplify the prototype into a system of point masses. The point masses of the two cylinders is placed along a two-dimensional line running through the geometric center of the cylinders. illustrated in figure 3.6.

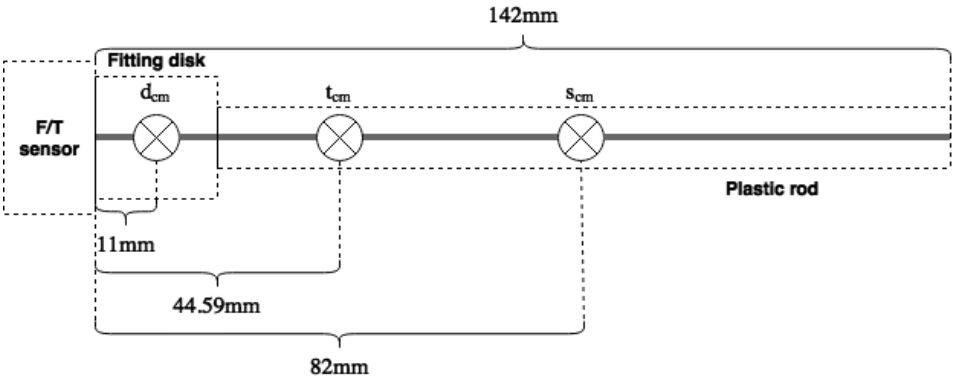


Figure 3.6: Point masses for the aluminium fitting disk d_{cm}, plastic rod s_{cm} and total center of mass t_{cm} for prototype 0.1 is highlighted.

The system consists of mass particles P_i, each with mass m_i that is located a distance x_i relative to the F/T sensor calibration point for $i = 1, \ldots, n$. The distance X to the center of mass can be derived from the following equation.

$$\sum_{i=1}^{n} m_i(x_i - X) = 0$$

Solving with regards to X yields the equation,

$$X = \frac{1}{M}\sum_{i=1}^{n} m_i x_i \tag{3.1}$$

where M is the sum of masses for all the particles. Inserted values presented in table 3.2 gives an estimate of the true center of mass for prototype 0.1.

$$X = \frac{m_1 x_1 + m_2 x_2}{m_1 + m_2} = \frac{98g \times 11mm + 88g \times 82mm}{98g \times 88g} = 44.59mm$$

Although the design of prototype 0.1 was very safe and enabled the initial experimental controller tuning, it was far removed from any practical implementations of a robotic rehabilitation system.

3.3.0.2 Prototype 1.0 and Prototype 1.1

The end-effector was redesigned after the external controller design solidified into a stable and safe version. The new design focused on a more practical and rehabilitation-oriented approach. The two main criteria in the design were to effectively transfer the weight of the stroke patients arm onto the robot and to avoid obstructing the stroke patients grasping functionality. Section 2.1.2 presents a recommendation that pose restrictions on the final prototype design. The restriction is that the final prototype design can not be based on mechanisms that lock or hinder movements in the stroke patients wrist. The restrictions seek to avoid implementing constraint-induced movement therapy in the stroke patient wrist for all future functionality provided by the robotic rehabilitation system.

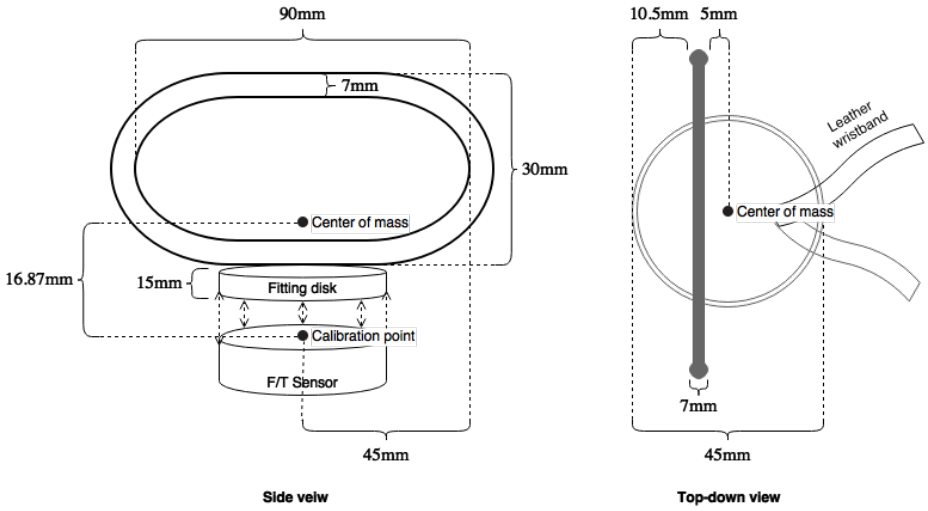


Figure 3.7: The final version of the hand splint interface is prototype 1.1 with an expanded aluminum ring and a leather wristband. The simplified center of mass is highlighted.

Prototype 1.0 incorporated an elliptically shaped hollow aluminum pipe that together with a rubber wristband is mounted to the sensor on a fitting disk crafted from plastic. The stroke patients hand would be inserted into the aluminum ring with the thumb outside the ring's elliptical perimeter. Thus, the aluminum ring would efficiently and accurately transfer human generated forces and torques onto the sensor without the need to grasp the ring. An exception arises if the stroke patients try to withdraw the hand with an open fist. A rubber band mounted to the fitting disk is intended to transfer the forces generated during this particular motion. The dimensions of the aluminum ring in Prototype 1.0 did not allow for a full insertion of the hand that went past the knuckles. In addition, the rubber wristband broke during testing. Improvements to the design are needed.

Prototype 1.1 was developed based on the previous prototype with some design changes. The changes include increasing the aluminum ring length by $20mm$, height by $5mm$ and replacing the rubber wristband with a leather wristband. The improved design allowed

for a full insertion of the hand past the knuckles and the strengthen wristband never failed during testing. Dimensions and details of prototype 1.1 are illustrated in figure 3.7.

Center of mass for Prototype 1.1

A good estimate of the center of mass ensured more accurate force/torque readings with less impact from the weight of the equipment upon the data readings. The method for calculating the center of gravity follows the procedure established in section 3.3.0.1 with some notable differences. The aluminum ring is mounted slightly offset from the fitting disk radius but it is counteracted by the mounting offset of the leather wristband. As a simplification, the weights of the aluminum ring and leather wristband is merged into a single point mass centered inside the aluminum ring and the fitting disk radius. The simplified communal point mass r_{cm} is located $30mm$ relative to the F/T sensor calibration point illustrated in figure 3.8.

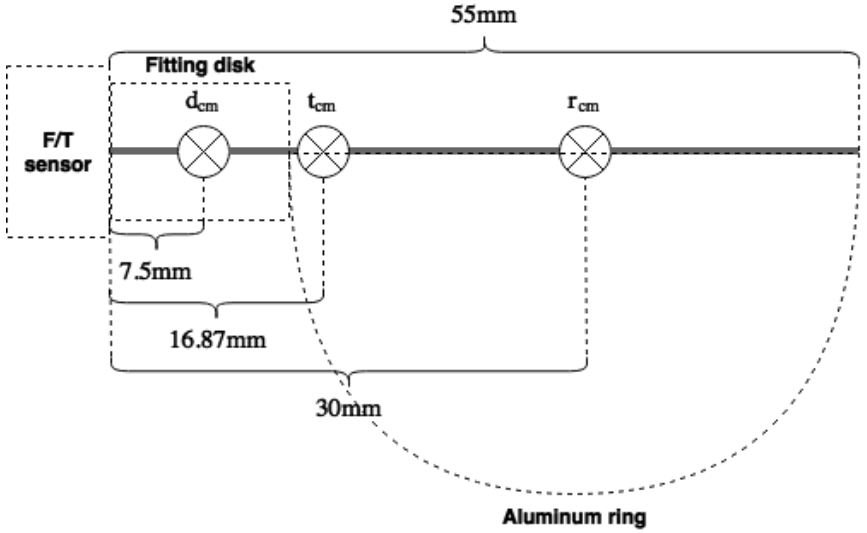


Figure 3.8: Point masses for the plastic fitting disk d_{cm}, aluminum ring and leather wristband r_{cm} and total center of mass t_{cm} for prototype 1.1 is highlighted.

The center of mass is calculated using equation 3.1 and values from table 3.2,

$$X = \frac{m_1 x_1 + m_2 x_2}{m_1 + m_2} = \frac{35g \times 7.5mm + (18+7)g \times 30mm}{25g \times 35g} = 16.87mm$$

It is important to note that the simplified center of gravity illustrated in figure 3.7 and figure 3.8 is not the true center of gravity, but the estimate is good enough for practical use in gravity compensation.

Weights of prototype elements

Table 3.2 presents the measured weights of all elements used in prototype 0.1 and 1.1. The measurement equipment operates with an accuracy of $\pm 1g$. An aluminum block with dimensions $5cm \times 4.9cm \times 1cm$ is utilized to perform a control weight. With a stated density value of $2.7g/cm^3$ the theoretical weight of the aluminum block is calculated to be $(5cm \times 4.9cm \times 1cm) \times 2.7g/cm^3 = 66.15g$. The measured weight of the aluminum block is $64g$, resulting in an experimental measurement error of 3.36% for the control weight. Some of the derivations can be attributed to inaccurate cutting and measuring the aluminum block dimensions and metal impurities in the aluminum. The overall assessment is that the weight utilized for these measurements is accurate enough for practical purposes.

Prototype 0.1 element	Weight [g]	Prototype 1.1 element	Weight [g]
aluminum fitting disk w/ screws	98	plastic fitting disk w/ screws	35
plastic rod	88	aluminum ring w/ screws	18
-	-	leather wristband w/ screws	7
Total w/ screws	186	**Total w/ screws**	60

Table 3.2: Weight of prototype elements. Measurements taken with a producer specified accuracy of $\pm 1g$ and experimentally derived accuracy of $\pm 3.36\%$.

3.4 External computer

The robotic rehabilitation system is implemented on an external computer running a Linux-based OS called Ubuntu 14.04. The program code is written in C++ in order to increase the real-time performance. A non-standard Linux library called GNU Scientific Library [26] is utilized for solving parts of the mathematical modeling.

3.5 Stationary structural hardware

The UR5 robot manipulator base is mounted $105cm$ above the floor on a heavy duty steel pedestal. The pedestal consists of a baseplate with dimensions $70 \times 70 \times 1.5cm$, a hollowed square pillar with rounded edges and dimensions $10 \times 10 \times 100cm$ and a circular mounting plate on the top and bottom. The circular mounting plates have a height of $1cm$ and $2.5cm$ respectively, and a shared radius of $\approx 7.48cm$ approximated from a measured circumference of $47cm$. Attached to the pedestal is a table top shaped as a quarter circle with inner radius $r_t = 40cm$ and outer radius of $R_t = 90cm$ mounted at a height of $72cm$ above the floor. The table is intended as a platform for interactions between the UR5 robot manipulator and the stroke patients. The pedestal and table are not bolted to the floor, but the weight of the heavy equipment in addition to the weight of the robot manipulator does not allow for much movement.

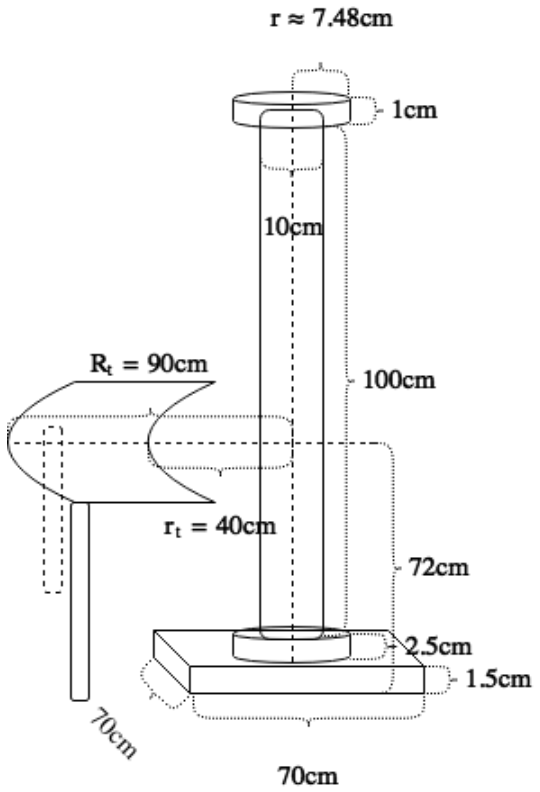


Figure 3.9: Illustration of the pedestal and table utilized together with the UR5 robot manipulator.

3.6 Complete system setup

The hardware and software components are now combined into the completed robot rehabilitation system utilized in this master thesis. An overview of the system architecture is illustrated in figure 3.10. The entire system is designed to be modular and interchangeable with minimal dependencies.

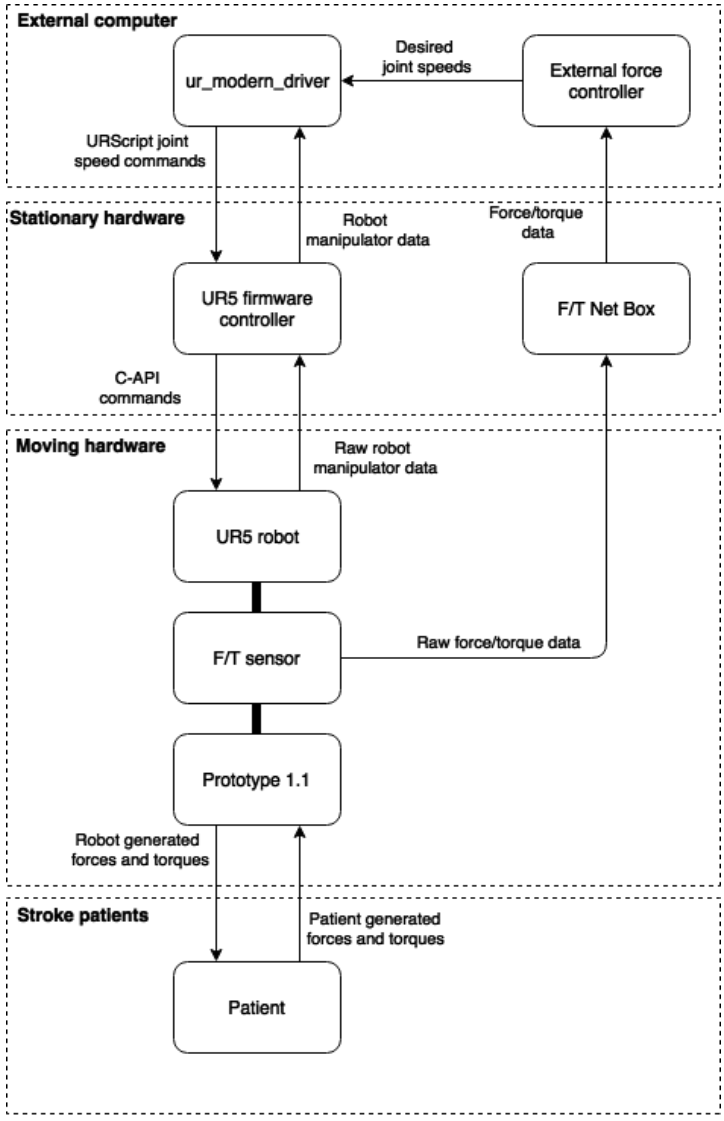


Figure 3.10: System architecture of the robotic rehabilitation system with critical interactions highlighted. All the main components in the modular design can easily be changed or replaced.

The robot rehabilitation system illustrated in figure 3.10 is implemented in the laboratory environment setup depicted in figure 3.11. The table-top is angled towards the workstation in order to quickly switch between code development and experimental testing. The UR5 Controller Box, F/T sensor and external computer form a small and isolated local network.

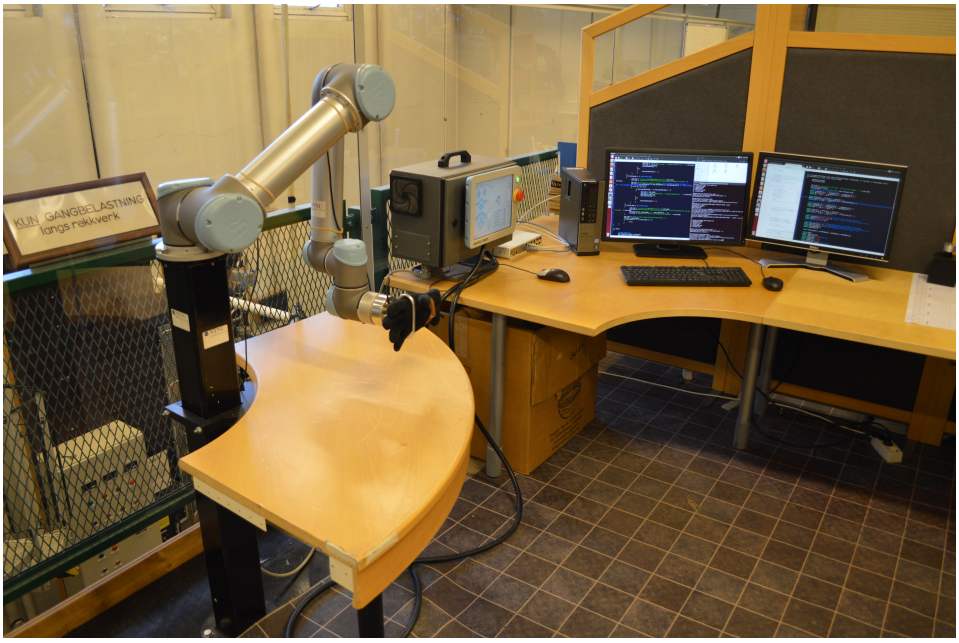


Figure 3.11: Completed system setup. The UR5 robot manipulator is mounted on the pedestal. Prototype 1.1 is mounted to the end-effector with an illustrative glove attached. In the background, the Controller Box with the teach pendant attached on the front together with a PoE switch and an external computer. The F/T Net Box is hidden from view behind the tabletop.

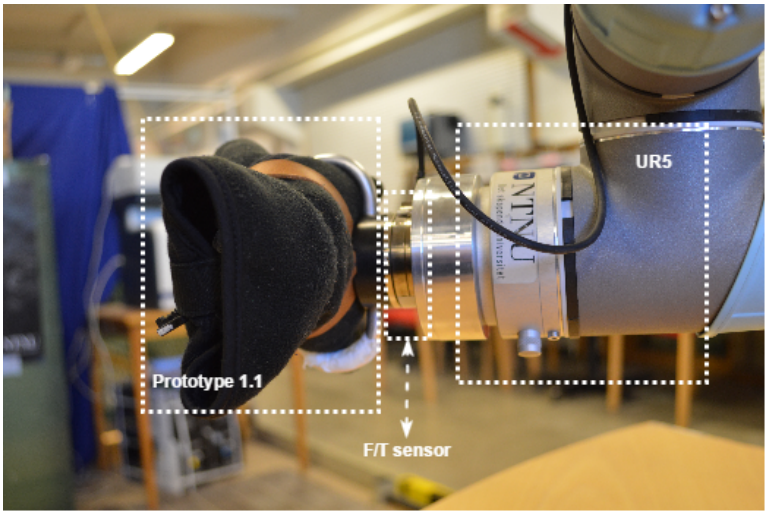
3.6.1 Laboratory safety

A risk assessment is performed in order to maintain the safety of the operator and other personnel during testing and development. One of the most important preventive safety measures described in the risk assessment is a safety checklist. The checklist consists of several steps designed to incrementally test the external controller behaviour as it is exposed to increasing levels of physical perturbations. The Controller Box and teach pendant is moved onto the worktable as a preventive security measure in the laboratory setup. This ensures easy access to the emergency stop button located on the teach pendant. Movement of personal inside the robot workspace is minimized through the manipulator's placement and mental awareness. Numerous of security measures limit the access to the laboratory, robot hardware and robot software. The entire risk assessment document can be found in Appendix C.

3.6.2 End-effector design

Two close-up pictures in figure 3.12 shows the correct insertion of the patient hand with an illustrative glove substitute. The same glove was also used during testing to ensure a thigh fit inside the aluminum ring in addition to providing some comfort to the operator.

(a) Close-up of Prototype 1.1 with an illustrative glove.



(b) Close-up of the end-effector with the main components highlighted.

Figure 3.12: Close-up pictures of the UR5 robot manipulator end-effector.

The robot Controller Box and the F/T Net Box is connected though a PoE switch to an external computer. The communication between the robot and external computer is es-

tablished though the ur_modern_driver that utilizes the real-time client interface with a maximum update frequency of $125Hz$. The F/T sensor broadcast data though the PoE switch using a UDP connection with a maximum sampling rate of $7000Hz$. A diagram showing the robot rehabilitation system architecture is illustrated in figure 3.10.

3.6.3 System delay

The laboratory setup introduces several delay sources that could impact the system performance. Streaming URScript speedj commands through the ur_modern_driver introduce an insignificant actuation delay between the target speed and actual speed[2]. The F/T Net Box delivers F/T data to its Ethernet port with a delay dependent on the utilized signal filtration[4]. Figure B.4b shows a delay of $286\mu s$ without any low-pass filtering enabled and a $36ms$ delay for low frequency ($> 1Hz$) signals exposed to low-pass filtration with a $5Hz$ cut-off frequency.

Transportation delay

The Ethernet connection between the UR5, F/T Net Box and external computer utilize socket communications based on TPC/IP and UDP/IP that introduce network delays to the system. The network delay can be tested through a simple pinging test. The suggested test involves pinging the sensor and robot from the external computer and recording the average round-trip time (RTT) for each data package. The ping command is normally only intended to confirm the existence of network components on a network and several sources caution about the accuracy of the RTT as a measure for the network delay. Several factors including package routing, package order, quality of service and the fact that the ping command utilizes the ICMP/IP message type instead of TCP/IP or UDP/IP can affect the accuracy of the network delay estimate. The impact of these factors is however insignificantly small for the practical implementation in the laboratory setup. Firstly, the only available network route for data packages is from the external computer, through the PoE switch and then into either the F/T sensor or UR5 Controller Box. Therefore, any RTT differences due to different package routes or package order are negated. Additionally, the PoE switch is more than capable of handling the amount of data traffic on the isolated local network. This negates the negative effects from switch buffering or differences in the quality of service. A switch will route IMCP and TCP/UDP packages slightly differently. This difference is more professionally interesting than relevant since it provides an almost identical network delay with modern network components. It is therefore assumed that utilizing the ping command for network delay testing is relevant, with results that will be a good network delay estimate for practical applications in the laboratory setup. The ping test results is presented in table 3.3.

UR5 Controller Box	Delay [ms]	F/T Net Box	Delay [ms]
RTT min	0.537	RTT min	0.541
RTT avg	0.590	RTT avg	0.589
RTT max	0.650	RTT max	0.658
RTT mdev	0.041	RTT mdev	0.032

Table 3.3: The F/T sensor and UR5 Controller Box is pinged 100 times (0% package loss) from the external computer.

The network transportation delay from the F/T sensor, through the external computer and into the UR5 Controller Box is estimated at,

$$\tau_{network} \approx \frac{RTTavg_{F/Tsensor}}{2} + \frac{RTTavg_{UR5}}{2} = 0.5895ms$$

The total system delay, from sensory input is detected to manipulator movement begins, can be estimated from the F/T Net Box filtration delay, network transportation delay, external controller iteration time and the speedj actuation delay.

$$\tau_{system} \approx \tau_{filtration} + \tau_{network} + \tau_{controller} + \tau_{actuation}$$

The actuation delay between the manipulators target speed and actual speed introduced through streaming the speedj command is insignificant, $\tau_{actuation} \approx 0s$, and is therefore excluded.

$$\tau_{system} \approx \tau_{filtration} + \tau_{network} + \tau_{controller} \tag{3.2}$$

In a best case scenario the system delay is estimated to,

$$\tau_{system} = 0.286ms + 0.5895ms + 0.008ms = 0.8835ms$$

and in a worst case scenario the system delay is estimated to,

$$\tau_{system} = 36ms + 0.5895ms + 0.008ms = 36.5975ms$$

The final estimate is dependant on the external controller frequency and the F/T sensor low-pass filtration delay.

3.7 Summary

Control method

The ur_modern_driver together with the URScript speedj command is utilized to ensure the best possible performance towards implementing indirect force control of the UR5 robot.

Force/Torque sensor

The six-axis Mini45-E force/torque sensor from ATI Industrial is utilized in the laboratory setup. The sensor data broadcast over a UDP connection through a PoE switch to an external computer. Several broadcasting frequencies and low pass filtration cut-off frequencies are made available through the F/T Net Box interface.

End-effector design

Three different prototypes were developed for use in this master thesis and are shown in figure 3.13. The final version is prototype 1.1 with a plastic fitting disk, aluminum ring and leather wristband illustrated in figure 3.13c. The total weight, including screws and some tape for comfort, adds up to $60g$ with a center of mass at an estimated distance of $16.87mm$ from the F/T sensor calibration point.

(a) Prototype 0.1 **(b)** Prototype 1.0 **(c)** Prototype 1.1

Figure 3.13: Pictures of the three different prototypes developed as part of this master thesis.

Safety

End-effector design, software driver, internal firmware controller and several safety measures implemented after a risk assessment ensure safety during testing and development of the external controller.

System delay

Information from technical documentation and a simple network transportation delay test can estimate the total delay from sensory input to movement begins in the manipulator. The estimate is dependant on the external controller frequency of and F/T sensor low-pass filtration delay.

Chapter 4

Manipulator modeling

The hardware and software foundations for the robotic rehabilitation system are established in the previous chapters. The feedback control loop is intended to operate directly in tool space to encourage a natural interaction between the robot manipulator and the stroke patient. Transforming the control signals between the tool space and the joint space presented by the velocity joint interface require a mathematical model of the of the robot manipulator. This chapter presents the kinematics and velocity kinematics of the UR5 robot manipulator in accordance with the established convention.

4.1 Kinematics

Kinematics is a geometric description of the manipulator motion obtained without considering the forces and torques that cause the motion. Kinematics are often divided into two parts; forward- and inverse kinematics. Forward kinematics are used to determine the position and orientation of the end-effector by utilizing the manipulator joint configuration. This is achieved by assigning a coordinate system to each joint and the corresponding links. Similarly, inverse kinematics is used to obtain the joint configuration based on the position and orientation of the end-effector. The preferred and standardized convention for robotic modeling is the Denavit-Hartenberg (DH) convention. Therefore, the DH convention will be utilized in this master thesis.

4.1.1 Forward kinematics

Forward kinematic equations define a set of transformation matrices between the joint space defined by the joints angular positions and the tool space defined by the end-effector position and orientation in Euclidean space. Any arbitrary motion by a rigid robot manipulator can be described by six parameters; three describing translation- and three describing

rotation. The resulting transformation matrix A_i is defined as;

$$A_i = \begin{bmatrix} R_i^{i-1} & o_i^{i-1} \\ 0 & 1 \end{bmatrix} \quad , i \in \{1, 2, \ldots, n\} \tag{4.1}$$

where i is the current joint number, R_i^{i-1} is the rotation matrix and o_i^{i-1} is the translation vector.

The DH convention can simplify the calculations of an arbitrary homogeneous transformation matrix by reducing the number of variables from six to four parameters. It is therefore used to determine the transformation matrices for the forward kinematics calculations. The method used for deriving the DH convention is presented in appendix B.1.1. The UR5 robot manipulator consists of six revolute joints with a joint range of $\pm 2\pi[rad]$. The DH parameters for the UR5 is presented in table 4.1.

Link	θ_i[rad]	d_i[mm]	a_i[mm]	α_i[rad]
1	q_1	89.459	0	$\frac{\pi}{2}$
2	q_2	0	-425	0
3	q_3	0	-392.25	0
4	q_4	109.15	0	$\frac{\pi}{2}$
5	q_5	94.65	0	$-\frac{\pi}{2}$
6	q_6	82.3	0	0

Table 4.1: Denavit-Hartenberg parameters for UR5

The transformation matrix A_i can now be expressed as a function of the four DH parameters θ_i, d_i, a_i and α_i for $i \in \{1, 2, \ldots, n\}$.

$$A_i = Rot_{z,\theta} Trans_{z,d} Trans_{x,a} Rot_{x,\alpha} \tag{4.2}$$

$$A_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.3}$$

The transformation matrices describing the position and orientation between each joint in the UR5 manipulator can be derived from equation 4.3. The six resulting transformation matrices, A_i for $i \in \{1, 2, \ldots, 6\}$, can be specific to the UR5 manipulator given the DH parameters presented in table 4.1.

A generic homogeneous transformation matrix T_i^{i-1} is given by,

$$T_j^i = \begin{cases} A_{i+1}A_{i+2}\ldots A_{j-1}A_j & \text{if } i < j \\ I & \text{if } i = j \\ (T_j^i)^{-1} & \text{if } i > j \end{cases}$$

The homogeneous transformation matrices T_j^0 describes the position and orientation of each joint frame $o_j x_j y_j z_j$ in relation to the base frame $o_0 x_0 y_0 z_0$.

$$\begin{aligned} T_1^0 &= A_1 \\ T_2^0 &= A_1 A_2 \\ T_3^0 &= A_1 A_2 A_3 \\ T_4^0 &= A_1 A_2 A_3 A_4 \\ T_5^0 &= A_1 A_2 A_3 A_4 A_5 \\ T_6^0 &= A_1 A_2 A_3 A_4 A_5 A_6 \end{aligned}$$

The final homogeneous transformation matrix T_6^0 is used extensively in the robotic rehabilitation system to transform control inputs from the base frame into the tool frame. Chapter 6 presents several reference vectors defined in the base frame that needs to be transformed into the tool frame before being applied as a control input to the external controller.

4.1.2 Inverse kinematics

Inverse kinematics is used to obtain the joint configuration based on the position and orientation of the end-effector. Most industrial manipulators are controlled on a low-level in joint space. Solving the inverse kinematic problem is generally more difficult than the forward kinematics problem. There are no guarantees that a solution exists, and if a solution exists it may or may not be unique. Uniqueness is an important issue as a single end-effector pose often can be realized by several joint configurations. The worst case scenario is a singularity defined as a specific end-effector pose that can be realized by an infinite number of joint configurations. Inverse kinematic solvers must handle non-unique solutions and singularities.

Complementary details about inverse kinematics can be found in [28]. The external controller presented in chapter 6 do not rely on inverse kinematics, but understanding the general method behind inverse kinematics is important as the internal controller might utilize an inverse kinematic solver.

4.2 Velocity kinematics

Velocity kinematics relates the linear and angular velocities of the end-effector with the joint velocities. The velocity relationship is determined by the Jacobian of the forward kinematics function. The Jacobian transforms between the joint velocities $\dot{q} \in \mathbb{R}^n$ and the linear- and angular velocities of the end-effector $\xi \in \mathbb{R}^6$. It is often divided into an analytic and geometric representation depending on the desired application. The analytic Jacobian, $J_a(q)$, is derived directly through differentiation of the forward kinematics. It is useful if the end-effector position and orientation is minimally represented through e.g. Euler angle parameterization. The geometric Jacobian, $J_g(q)$, is derived through geometrical properties where each joint velocity is related to changes in the linear and angular velocities of the end-effector. The two representations are mathematically connected,

$$J_a(q) = \begin{bmatrix} I & 0 \\ 0 & B^{-1}(\alpha) \end{bmatrix} J_b(q) \quad \text{, provided that } det(B(\alpha)) \neq 0$$

with Euler angles $\alpha = [\phi, \theta, \psi]$ and $B(\alpha) = \begin{bmatrix} \cos\psi \sin\theta & -\sin\psi & 0 \\ \sin\psi \sin\theta & \cos\psi & 0 \\ \cos\theta & 0 & 1 \end{bmatrix}$.

This master thesis is mainly concerned with the velocity relationship between the task frame and joint space. Therefore, the geometric Jacobian is the most relevant representation.

4.2.1 Geometric Jacobian

The geometric Jacobian is normally divided into a linear- and angular velocity partition. Determining the partitions $J_{v,i}$ and $J_{\omega,i}$ is based on whether the i^{th} joint is revolute or prismatic.

$$J_i = \begin{bmatrix} J_{v,i} \\ J_{\omega,i} \end{bmatrix} = \begin{cases} \begin{bmatrix} z_{i-1}^0 \times (o_n^0 - o_{i-1}^0) \\ z_{i-1} \end{bmatrix}, & \text{if joint } i \text{ is revolute} \\ \begin{bmatrix} z_{i-1} \\ 0 \end{bmatrix}, & \text{if joint } i \text{ is prismatic} \end{cases} \tag{4.4}$$

The UR5 manipulator consists of six revolute joints , therefore J_i can be defined as,

$$J_i = \begin{bmatrix} J_{v,i} \\ J_{\omega,i} \end{bmatrix} = \begin{bmatrix} z_{i-1}^0 \times (o_n^0 - o_{i-1}^0) \\ z_{i-1} \end{bmatrix}, \quad i \in \{1, 2, \ldots, 6\}$$

4.2.2 Manipulator Jacobian

The manipulator Jacobian related the vector of joint velocities to the body velocity $\xi = \begin{bmatrix} v^T & \omega^T \end{bmatrix}^T$ of the end effector. The skew matrix $S(\omega(t))$ can be used to determine the

angular velocity vector ω_n^0 partition of the manipulator Jacobian. It is defined by the rotational matrix R_i^{i-1} from the transformation matrix A_i in equation 4.1.

$$S(\omega_n^0) = \dot{R}_n^0 (R_n^0)^T = \begin{bmatrix} 0 & -\omega_{n,z}^0 & \omega_{n,y}^0 \\ \omega_{n,z}^0 & 0 & -\omega_{n,x}^0 \\ -\omega_{n,y}^0 & \omega_{n,x}^0 & 0 \end{bmatrix} \tag{4.5}$$

The linear end-effector velocity vector υ_n^0 can be determined directly from

$$\upsilon_n^0 = \dot{o}_n^0 \tag{4.6}$$

The manipulator Jacobian $J \in \mathbb{R}^{6\times n}$ can now be defined by the linear- and angular partitions,

$$\upsilon_n^0 = J_\upsilon \dot{q} \tag{4.7}$$

$$\omega_n^0 = J_\omega \dot{q} \tag{4.8}$$

where $J_\upsilon, J_\omega \in \mathbb{R}^{3\times n}$. Finally, the manipulator Jacobian can be expressed as,

$$\xi = J\dot{q} \tag{4.9}$$

where $\xi = \begin{bmatrix} \upsilon_n^0 \\ \omega_n^0 \end{bmatrix}$ and $J = \begin{bmatrix} J_\upsilon \\ J_\omega \end{bmatrix}$.

4.2.3 Inverse velocity kinematics

The intended low-level control method in this master thesis is to control the UR5 manipulator through joint velocity control inputs. Therefore, calculating the inverse velocity kinematics become relevant. Assuming that the Jacobian is square and non-singular the inverse velocity problem can be solved by the inversion of the Jacobian in equation 4.9.

$$\dot{q} = J^{-1}\xi \tag{4.10}$$

The manipulator Jacobian will always be square since the UR5 consists of exactly six revolute joints, resulting in a $J \in \mathbb{R}^{6\times 6}$ manipulator Jacobian. A singular solution is defined by a configuration at which the Jacobian loses rank, more precisely any configuration q such that $rank[J] \leq max_q(rank[J(q)])$. Avoiding singularities is therefore crucial in order to calculate the inverse velocity kinematics. The robotic rehabilitation system is implemented with a non-standard Linux library to assist with these calculations.

4.2.4 Force control

The Jacobian can also be used to relate forces F applied at the end-effector tool frame to the induced joint torques τ.

$$\tau = J^T(q)F \tag{4.11}$$

Direct control of the joint torques in the UR5 robot manipulator is unfortunately not supported by Universal Robots. This lack of support for low-level joint torque control is presented and discussed in chapter 3.

4.3 Summary

The robotic rehabilitation system is intended to be implemented on an external controller with a control feedback loop operating directly in the tool frame.

The forward kinematics for the UR5 manipulator is derived based on the DH convention supplied by Universal Robotics. The resulting homogeneous transformation matrix T_6^0 can be utilized to transform reference vectors defined in the base frame into the tool frame.

The geometric Jacobian is derived and presented together with the manipulator Jacobian. The inverse velocity kinematics is derived from the manipulator Jacobian. The resulting equation 4.10 can be utilized to realize the desired joint velocity control interface.

Chapter 5

Stroke patients

The safe interaction between stroke patients and an industrial robot manipulator is a crucial condition for this master thesis. This chapter contains several implementation challenges directly related to the safety of stroke patients utilizing the robotic rehabilitation system. Firstly, identifying and choosing a suitable sampling frequency is important to properly capture the delicate interaction through the external F/T sensor. Three implementation challenges are highlighted in this chapter, as they are relevant to the delicate interaction between the UR5 robot manipulator and stroke patients. The implementation challenges is identified, analyzed and discussed before the implemented solutions are analyzed for potential consequences to the performance of the robotic rehabilitation system. Finally, an overview of the safety mechanisms provided by the robotic rehabilitation system is presented.

5.1 Sampling frequency

The sensor sampling frequency is crucial in order to properly capture the delicate interaction between the robot manipulator and the stroke patients. Identifying the frequency of the quickest dynamic involved in the robotic rehabilitation system is important in order to establish a suitable sampling frequency. Additionally, it needs to be established whether any symptoms in stroke patients pose an extended challenge to the data sampling. Stroke patients may suffer from a large variety of complications depending on the size and location of the affected area in the brain. Spasticity and trembles are among the possible complications after a stroke that can generate rapid movements in the patients. There is a significant number of other complications stroke patients may suffer that is not covered in this section. However, none of them pose a significant challenge to the data sampling frequency.

The sampling frequency is estimated based on Nyquist's sampling theorem. The theorem states that a theoretically minimal sampling frequency should be at least twice as fast as the quickest dynamic in the system.

Spasticity

Spastic movements are a common symptom after a stroke and are characterized by excessive velocity-dependent muscle contraction that ultimately leads to hyperreflexia, an exaggerated deep tendon reflex. Detecting a sudden change in the velocity in the affected extremities triggers the spastic reflex. The detection relies on the human nervous system. Therefore, it is safe to assume that any spastic movement will have a dynamic rate equal, or lower than, the quickest human reaction time.

Tremors

Although tremors are a fairly uncommon symptom in stroke patients it is important to investigate in the pursuit of establishing the quickest dynamics in the system. Tremors are defined as repetitive, oscillatory movements caused by alternate, or synchronous, but irregular contractions of opposing muscle groups. The tremor movements are usually involuntary and can be classified into three main categories low-, middle- and high-frequency tremors. The tremor movements generated is defined by the lowest category as tremor frequencies less than $4Hz$, the middle category is tremor frequencies between 4 and $7Hz$ and the highest category is tremor frequencies higher than $7Hz$ [1].

Human movements

Generally, human motion is defined as a series of nerve commands communicated to a specific muscle. In order to maintain a constant muscle tension, the nerves maintain a maximum nerve signal frequency of $\approx 200Hz \rightarrow 300Hz$ limited by the nerve's action potential update frequency. However, this dynamic is not relevant as it occurs on a physiological scale that is too small to be captured in the laboratory setup. Movements that occur on a larger human scale, both voluntary and involuntary, is more relevant for this practical application.

According to Gertjan Ettema Ph.D., a professor at Department of Neuroscience, Faculty of Medicine NTNU, the quickest human reaction time is approximate $25ms$. The reaction time is measured from the initial human sense input is registered by the central nervous system, to movement begins in the extremities. This insight was gained during a course lecture[8] in October 2016 and subsequent discussions with the professor. None of the symptoms discussed produce a motion frequency higher than the quickest human reaction time. It is safe to assume that all other voluntary movements have a dynamic rate slower than the quickest human reaction time. Assuming that the quickest human dynamic is defined by a repetitive motion based on the quickest human reaction time is a worst case scenario.

$$f_{human} = \frac{1}{25ms} = 40Hz$$

Robotic manipulator movements

The hardware utilized in this thesis is capable of running at high frequencies. The UR5 robot manipulator is hardware limited to a maximum joint servo updating rate of $125Hz$,

and the F/T sensor can sample data at a maximum frequency of $7000Hz$. In the interest of maximizing the potential of the robotic rehabilitation system, it is proposed that the external controller should run at the maximum updating frequency of $125Hz$. Assuming that the quickest robot dynamic is defined by a repetitive motion based on the maximum joint servo updating rate is a worst case scenario.

$$f_{robot} = 125Hz$$

F/T sensor sampling frequency

The analysis presented in section 5.1 show that $f_{human} < f_{robot}$. The quickest dynamic in the robotic rehabilitation system is therefore defined by the quickest robot dynamic. The minimal theoretical sensor sampling frequency needed to sample the robotic rehabilitation system can be estimated with Nyquist's sampling theorem.

$$f_{sampling} = f_{robot} \times 2 = 125Hz \times 2 = 250Hz$$

The suggested sensor sampling rate of $250Hz$ is more than six times faster than the highest dynamic rate expected to be produced by human motion. This might seem excessive, but Nyquist's sampling theorem only provide a theoretical minimal sampling frequency. The norm in practical applications is often to oversample the signal by several factors of the Nyquist rate. Oversampling generally produces a better quality signal with improved resolution and reduced noise.

5.2 Implementation challenges

Three implementation challenges are highlighted in this section, as they are relevant to the delicate interaction between the UR5 robot manipulator and stroke patients. The three implementation challenges are identified, analyzed and discussed before the implemented solutions are analyzed for potential consequences to the performance of the robotic rehabilitation system.

5.2.1 Unwanted oscillations

Observations from early testing and development revealed several implementation challenges with unwanted movements in the robot manipulator. The unwanted movements are generated from several sources, but can generally be identified as high-frequency oscillations. Regardless of the source the general behaviour caused by the unwanted oscillations is not directly dangerous for the patients, but can easily be perceived as uncomfortable and unnerving.

Human arm inertia

The unwanted oscillations connected with the inertia of a human arm is generated by an unintended and unwanted control feedback loop between the external controller and the

human arm. The unwanted behaviour is caused by rapid movement in the robot manipulator generating a delayed physical movement in the human arm due to inertia. The external controller then perceives that the arm is not directly following the manipulator's movements and produce a counter-movement in the manipulator for the next control iteration. The momentum in the human arm has now had time to build up before being met by the external controllers counter-movement. These interactions continue in the unwanted feedback loop resulting in unwanted oscillations with a high frequency in the end-effector.

Stroke patient symptoms

The tremors and spastic movements described in section 5.1 contribute to the unwanted oscillations. Observations from simulated tremor- and spastic test show that the associated movements often trigger the unwanted control feedback loop described earlier. The difference is that tremor movements can independently generate and maintain an unwanted control feedback loop through similar mechanics.

Analysis

Implementing control signal filtration able to differentiate between the patient generated movements defined as control feedback inputs and the patient generated movements defined as noise or random values is a challenging endeavor. The nonlinearities connected with stroke patient movement and the large variation in documented stroke symptoms place designing an accurate mathematical model outside the scope of this master thesis. The simplified approach in this master thesis constitutes differentiating the control signals by the frequency of the movement. It is reasonable to assume that high-frequency movements can be regarded as unwanted behaviour. Patients suffering from high-frequency trembles is a good example. The trembles do not aid the patient in performing a movement or task and is consistently defined as unwanted behaviour by a professional physical therapist. The suggested simplification to the control signal differentiation is substantial, but it is in line with the goals and objectives for the master thesis.

Removing unwanted high-frequency movements can easily be realized by implementing a low-pass filter. The low-pass filter will attenuate control signal with a frequency that exceeds a predetermined cut-off frequency. This reduces the unwanted oscillations in the robotic rehabilitation system and additionally attenuate the signals generated by the middle- to high-frequency tremors described in section 5.1. The counterargument is that a low-pass filter will attenuate conscious high-frequency movements generated by the patient.

Low-pass filtration

The F/T sensor provides a built-in low-pass filter with several predefined cut-off frequencies. Numerous cut-off frequencies are tested, but in the interest of maximizing the system's safety and stability the lowest available cut-off frequency of $5Hz$ is chosen. The low-pass filter attenuates control signals with a frequency $> 5Hz$ defined by the logarithmic behaviour in Appendix B.4a, and introduces a significant increase to the filtration

delay defined by appendix B.4b. It is possible that this implementation choice ignores voluntary patient generated motions, but it is a reasonable simplification. For practical use, it is highly unlikely that rehabilitation methods will introduce motions or exercises that require a higher movement frequency. To put it in context, health individuals will struggle to scribble on a piece of paper with a frequency equal to or higher than $5Hz$. The increased system delay estimated by equation 3.2 is seen a bigger challenge. Utilizing the lowest cut-off frequency produces a worst case scenario where the system delay is estimated to be $\tau_{system} = 36.5975ms$. This implies that the external controller might not have time to react to the quickest human movements defined by the human reaction time estimated to be $\approx 25ms$. Regardless of the discussed decrease in system performance, the increase in overall security in the robotic rehabilitation system takes precedence.

5.2.2 Mounting bias

Initial implementation of the robotic rehabilitation system reveled a constant bias in the raw F/T data that could generate unwanted movements in the end-effector. The F/T sensor bias is recorded after a clean restart of the F/T Net Box and presented in table 5.1.

Forces [N]			Torques [Nm]		
F_x	F_y	F_z	T_x	T_y	T_z
-19.59	-13.01	7.0862	-0.3561	0.66333	-0.225

Table 5.1: Arbitrary bias in F/T sensor without a F/T Net Box software bias enabled.

The arbitrary magnitude of the forces presented in table 5.1 would generate a substantial control input to the external controller that could pose a danger to the stroke patients. The consequences of a movement generated by the mounting bias is assessed to be small, with injuries that might require medical treatment. The probability of a movement generated by the mounting bias is however assessed to be very large, with weekly incidents. The risk assessment of the mounting bias situation requires preventive actions to be made to reduce the risk.

F/T sensor mounting bias

The Mini45 F/T sensor is mounted on the end-effector and prototype 1.1 with mounting screws illustrated in figure 3.12. It is speculated that these screws produce the constant mounting bias in the raw F/T data.

Analysis

The external controller can easily be designed to remove the mounting bias with a software bias. The F/T Net Box interface also enables the user to remove the mounting bias through a software bias. Implementing the external controller software bias is assessed to

only reduce the probability of the situation since a fault in the software bias fault still generates a control signal without a reduction in magnitude. Implementing the F/T Net Box software bias method is assessed to only reduce the consequence of the situation, since this method masks both the mounting bias and the gravity components of the equipment mounted to the end-effector. The equipment weight is masked in an initial tool frame, any rotational movement in the end-effector deviating from the initial tool frame will reveal the effect of the equipment weight on the F/T sensor readings. Continued movements in the manipulator after the patient releases or is not in contact with the end-effector prototype is unwanted. This implies that the F/T Net Box software bias reduces the consequence by masking the magnitude of the movements generated, but it does not reduce the probability of unwanted movements. Regardless of the discussed decrease in system performance, the increase in overall security in the robotic rehabilitation system takes precedence.

Double software bias

Based on the risk assessment the decision is to implement both methods for dealing with the mounting bias as a double preventive security measure. The consequences of this decision on the system performance need to be evaluated. The initial software bias in the F/T Net Box do not affect the filtration delay, and the external controller frequency is unaffected by introducing a software bias. The gravity compensation will suffer substantially from the masking of the gravity components. This effect will be most prevalent for a software bias initiated were the entire gravity vector is contained within a single axis. The measured weight- and estimated center of gravity in Prototype 1.1 is presented in table 3.2 and figure 3.8 respectively. These values can be utilized to calculate the maximum gravitational component of prototype 1.1 assuming an initial tool frame where the force and torque components are perfectly isolated and only acting on the associated axis.

$$F_{equipment} = m \times a \approx 0.06kg \times 9.81\frac{m}{s^2} = 0.5886N$$

$$\tau_{equipment} = F \times l \approx 0.5886N \times 0.01689m = 0.009941454Nm$$

The consequences to the worst case scenario with the gravity compensation can be negated with a simple dead-band filter. The dead-band filter is implemented on the force and torque error updates in the external controller with intervals $\pm 1N$ and $\pm 0.5Nm$. This prevents the external controller from drifting due to the compromised gravity compensation. The dead-band filter generally lowers the sensitivity of the robotic rehabilitation system, but it removed unwanted drifting in the end-effector. The general behaviour caused by the drifting is not directly dangerous for the patients, but can easily be perceived as uncomfortable and unnerving. Removing the unwanted drifting after the stroke patients release the end-effector reinforce the perception that they are in control over the situation. This is especially important during early introductions of the robotic rehabilitation system to new stroke patients.

The consequences and practical solutions to the mounting bias challenge negatively impacts the delicate interactions between the stroke patients and manipulator. The dead-band filter generally lowers the sensitivity of the robotic rehabilitation system, but removing the drift caused by a non-perfect gravity compensation is seen as more important.

5.2.3 Workspace limitations

Observations from early testing and development reveal rapid movements generated in the large base, shoulder and elbow joint when moving close to a singularity or the periphery of the UR5 workspace. These movements are dangerous to stroke patients due to the large momentum involved with large velocities in the three biggest revolute joints. The potential consequences from a self-collisions or crushing a hand between the larger joints are substantial. The probability of this occurring is moderate to high as the stroke patients are intended to move freely during rehabilitation exercises only restricted by the UR5 workspace and small table. This situation is defined in the risk assessment presented in Appendix C as the most dangerous collision between the UR5 manipulator and personnel, with the potential for serious injury with a moderate to high probability.

Singularities

The most dangerous situations occur while moving close to the cylindrical volume directly above and directly below the robot base illustrated in figure 5.1. The cylindrical volume is implemented to avoid a singularity but in order to avoid the volume, the manipulator must sometimes perform movements that induce large velocity fluctuations in the large base and shoulder joints. This behaviour poses a crush danger to the stroke patients through trapping a hand between the two links connecting the larger joints or self-collisions with the larger joints while they move at a high velocity. These high-velocity movements do not directly translate to changes in the end-effector movements but it is plausible it could trigger a spastic movement, regardless it is perceived as dangerous and should be avoided.

Inverse Jacobian

The inverse Jacobian solver utilized in this master thesis is speculated to be the source of some of the unwanted behaviour. A specific scenario occurs after reaching the edge of the firmware defined workspace while still being told by the external controller to maintain the current trajectory. This induces oscillations in the manipulator elbow joint while the high-speed external controller searches for a non-existent solution outside the predefined workspace. This behaviour is not directly dangerous for the patients, but can easily be perceived as uncomfortable and unnerving.

Analysis

The high-risk assessment requires preventive actions, in this master thesis, the practical solution is restricting the UR5 manipulator workspace to avoid the areas that cause the undesired behaviour.

The original workspace of the UR5 manipulator consists of an approximate sphere with a hollow cylindrical volume that is centered in the robot base. The workspace is illustrated in figure 5.1. The latest 3^{rd} generation internal controller enables the user to configure the safety parameters inside the Polyscope GUI. This includes easily modifying the workspace of the robot by limiting the TCP position. Sadly, this is not an option since the UR5 utilized in this master thesis runs an older 2^{nd} generation internal controller that does not support

user-generated changes to the safety configurations. It is evident that the UR5 workspace must be restricted by the external controller.

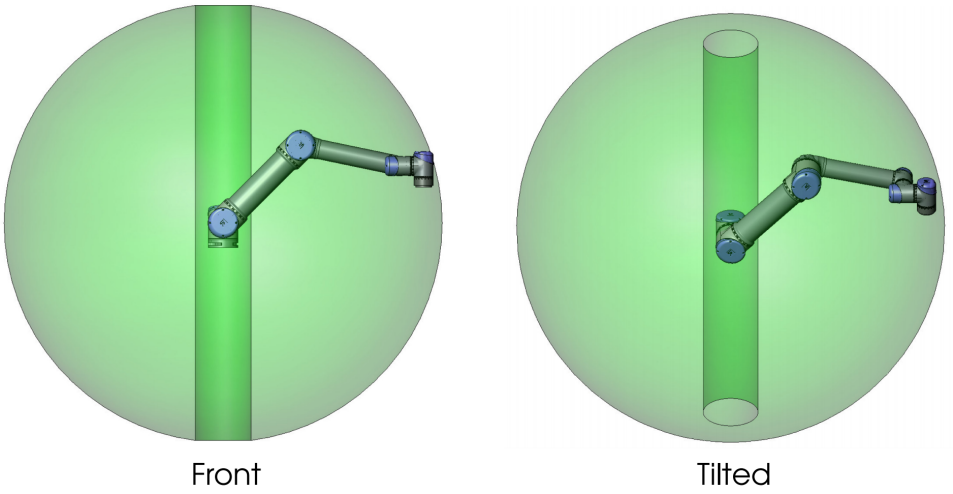


Figure 5.1: The original workspace of the UR5 manipulator consists of an approximate sphere with radius of $850mm$, except for a cylindrical volume with radius of $149mm$ directly above and directly below the robot base. Figure taken from [22].

Virtual workspace

Implementing an external controller enables the ability to restrict the original workspace into a smaller and simplified virtual workspace that reduce the probability of the unwanted behaviour. The virtual workspace boundaries consist of a large cylinder with a radius of $700mm$ and a smaller cylinder with a radius of $350mm$. Both cylinders have a height of $520mm$ that is centered $40mm$ above the base frame due to the mounting height of the small table. The reasoning behind making the virtual workspace cylindrical and not spherical is to encourage a natural interaction with the table. The entire virtual workspace is designed to fit inside the firmware defined workspace and still allow for movements and tasks to easily be performed inside the virtual workspace. The easiest implementation would be a hard boundary that triggered an emergency shutdown if the manipulator moved outside the virtual workspace. This implementation would be safe, but it is not suited in a practical implementation as the patients have no visual references of the virtual workspace boundaries. The virtual workspace is therefore implemented as a series of soft boundaries, were movements outside the virtual workspace is allowed but discouraged. The intended behaviour is that the external controller begins to resist the movements if the manipulator moves outside the virtual workspace. This is achieved through creating a counter-force vector with a direction pointed back towards the virtual workspace and an increasing magnitude that scales with the distance from the soft boundary. The virtual workspace is illustrated in figure 5.2.

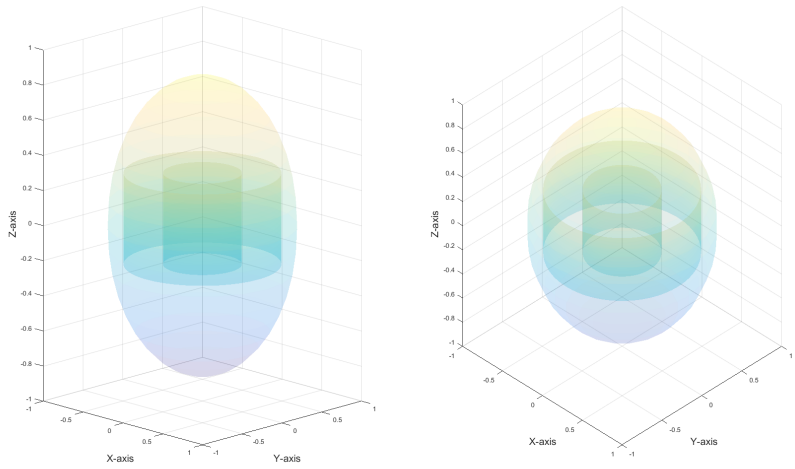


Figure 5.2: Virtual workspace defined by an inner- and outer cylinder centered $40mm$ above the base frame relative to the UR5 spherical workspace. Dimensions for height and radius of the inner- and outer cylinder is $520mm \times 350mm$ and $520mm \times 700mm$ respectively.

5.3 Executive controller security

The robotic rehabilitation system behaviour is defined by an internal firmware controller and an external controller. The executive controller design is presented in chapter 6. This section presents an overview of the internal- and external safety mechanisms. The safety mechanisms not previously presented by the implementation challenges is briefly described.

General emergency shutdown procedure

Events interpreted as dangerous to the stroke patients is handled by an implementation strategy common to both the internal- and external controller. These events trigger an emergency shutdown procedure were an emergency shutdown command is sent to the UR5 robot that immediately halts all movements in the manipulator. The command can be sent by the external controller through the ur_modern_driver to the Controller Box firmware, triggered by the user through the emergency button on the teach pendant or be issued by the internal firmware controller itself. The documentation specifies that all movement will be completely halted within $500ms$ given a worst-case scenario with a maximum TCP speed of $\approx 1m/s$ and the manipulator fully expanded horizontally [22].

5.3.1 Internal controller safety mechanisms

There are several limiting safety-related functions [23] built into the UR5 internal firmware controller that is constantly active while the robot is running. These safety functions in-

clude limiting all joints with a maximum joint range of $\pm 360^\circ$ and a maximum joint speed of $\pm 180^\circ/s$. Additionally, an emergency shutdown [25] is triggered when a maximum force acting on the robot Tool Center Point(TCP) exceeds $150N$, or a maximum momentum of the robot arm exceeds $25\frac{kg \times m}{s}$. There is some measurement uncertainty related to the robot's internal sensors. The producer specifies a force trueness of $25N$ and a momentum trueness of $3\frac{kg \times m}{s}$. The internal controller safety mechanisms is presented in table 5.2.

Limiting safety function	Description	Limit
Joint position	Min. and Max. angular joint position	$\pm 360^\circ$
Joint speed	Max. angular joint speed	$180^\circ/s$
TCP position	Sphere in Cartesian space limiting TCP position	$850mm$
TCP speed	Max. speed of the robot TCP	$\approx 1\ m/s$
Force	Max. pushing force of the robot	$150N$
Momentum	Max. momentum of the robot arm	$25\frac{kg \times m}{s}$
Power	Max. applied robot arm power	$300W$

Table 5.2: Firmware limiting safety-related functions.

5.3.2 External controller security mechanisms

Designing and implementing an external controller easily enables the ability to add extra security measures to the robotic rehabilitation system. Several added security measures trigger the general emergency shutdown procedure, while other security measures implement a soft boundary approach to encourage a more natural interaction. The predefined maximum levels are calibrated from what a healthy male of average height and weight find comfortable to handle. The control signal filtration and workspace limitations on the UR5 manipulator is presented in section 5.2.1/5.2.2 and 5.2.3 respectively. The external controller security mechanisms not presented and discussed by a specific implementation challenge is briefly presented. The external controller safety mechanisms is presented in table 5.3.

Limiting safety function	Description	Limit
TCP force	Max. pushing force on the robot TCP	$50N$
TCP torque	Max. rotating torque on the robot TCP	$8Nm$
TCP position	Cylinders generate virtual workspace for TCP	workspace
TCP position	Soft min. Cylinder height×radius [mm]	520×350
TCP position	Soft max. Cylinder height×radius [mm]	520×700
Time	Max. controller run-time	$200s$
F_m	Low-pass filter	$5Hz$
τ_m	Low-pass filter	$5Hz$
e_F	Dead-band filter	$\forall F_m \pm 1N$
e_T	Dead-band filter	$\forall \tau_m \pm 0.5Nm$
u_F	Max. controller force output	$\forall u_F \pm 5$
u_T	Max. controller torque output	$\forall u_T \pm 5$

Table 5.3: External controller limiting safety-related functions.

The external controller will timeout and shutdown after $200s$ to prevent accidental activation by the audience if the system is left running without supervision. The force- and torque data from the F/T sensor triggers an emergency shutdown if the readings exceed a cautiously low safety limit of $50N$ and $8Nm$ respectively. New errors or references that is passed to the external controller is incrementally increased with a gentle step-up or step-down in magnitude to prevent a stuttering behaviour and sudden movements in the manipulator. This includes normal force- and torque control inputs and the random force vectors generated in random mode, a high-level control strategy presented and described later in chapter 6. The controller outputs is limited for both forces and torques to $u_F, u_T < 5$, exceeding the output level triggers an emergency shutdown.

5.4 Summary

The external controller frequency is set to the maximum frequency of $125Hz$ to maximize the potential of the robotic rehabilitation system. The minimal required F/T sensor sampling frequency is therefore set to $250Hz$ in accordance with Nyquist's theorem.

The low-pass filtration is enabled in the F/T sensor to prevent unwanted oscillations between the patient and the manipulator. The cut-off frequency is set to $5Hz$, introducing a significant increase in system delay and possible loss of useful control signal in order to increase the overall security of the robotic rehabilitation system.

The double software bias is enabled in the F/T sensor and the external controller to prevent unwanted movements between the patient and the manipulator due to the F/T sensor mounting bias. The F/T sensor software bias mask the equipment's gravity components. The external controller dead-band filters the error updates based on a worst case scenario. The dead-band intervals is $\forall F_m = \pm 1N$ and $\forall \tau_m = \pm 0.5Nm$. This does not increase the system delay but the sensitivity of the system is negatively affected in order to increase the overall security of the robotic rehabilitation system.

The workspace of the UR5 manipulator is limited within a cylindrical band defined by an inner- and outer cylinder centered $40mm$ above the base frame. Dimensions for height and radius of the inner- and the outer cylinder is $520mm \times 350mm$ and $520mm \times 700mm$ respectively. The virtual workspace severely limits the UR5 manipulator workspace in order to increase the overall security of the robotic rehabilitation system.

The internal controller firmware safety mechanisms is presented and discussed. Numerous external security mechanisms are implemented in the external controller in order to increase the patients perceived and genuine security level. This includes several emergency shutdown procedures, soft boundaries and a virtual workspace.

Chapter 6

Controller design

The foundation for the robotic rehabilitation system is established through the rehabilitation strategies, laboratory setup, mathematical model and stroke patient oriented security measures. The components of the robotic rehabilitation system needs to be realized through the design and implementation of an external controller guided by the tasks and subtasks given by the master thesis assignment.

6.1 Gravity compensation

The ability to compensate for the weight of the end-effector equipment is important for an external controller based on force and torque control. Chapter 3 presents an estimate for the center of mass for end-effector prototype 1.0 and prototype 1.1. The estimated center of mass is utilized together with the measured weight of the equipment presented in table 3.2 and the forward kinematics equations presented in chapter 4. This enables the external controller to predict and compensate for the gravity vector generated by end-effector equipment.

The gravity compensation is compromised by the F/T software bias presented in section 5.2.2 that mask the gravity readings in the initial tool frame. To counteract the negative effects, an artificial gravity vector is introduced to separate the static F/T mounting bias and the dynamic gravity components of the equipment weight. The initial separation will hopefully enable the external controller to perform gravity compensation without the negative effects from the F/T software bias.

Experimental verification

The performance of the gravity compensation is experimentally verified in three experimental verification tests, one rotational test for each axis. The three tests manually rotate the end-effector around the relevant axis in two 90 degree increments for a total of 180

degrees. Each test is initiated with the tool frame in an identical position and orientation. After each rotation increment, there are no interactions between the operator and the end-effector. Force readings with the gravity compensation enabled are documented and plotted in the tool frame. A successful gravity compensation is expected to produce zero force readings in all three axes after each rotation increment. The results from the rotation test is shown in figure 6.1.

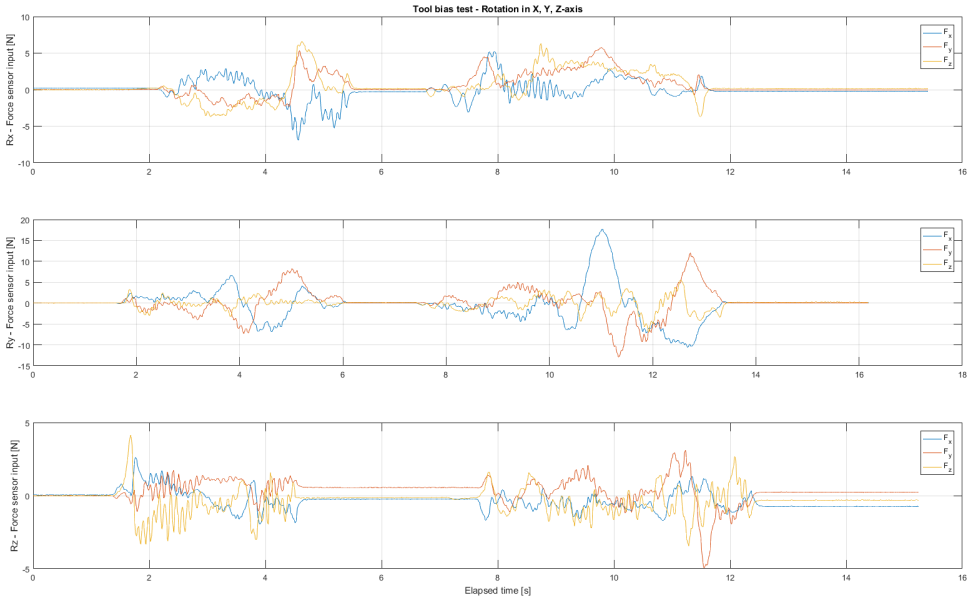


Figure 6.1: Gravity compensation test with a 180° rotation in two 90° increments around all three axes. All force readings are relative to the tool frame. Manual quarter rotation increment performed around the X-axis between $2.1s \rightarrow 5.6s$ and $6.8s \rightarrow 11.7s$, the Y-axis between $1.7s \rightarrow 6.0s$ and $7.5s \rightarrow 13.5s$ and the Z-axis between $1.3s \rightarrow 4.7s$ and $7.6s \rightarrow 12.5s$.

Performance analysis

The results illustrated in figure 6.1 shown an almost perfect gravity compensation with rotations abound the X- and Y-axis. The large spikes in force readings during each rotation increment is irrelevant for this test. There are some deviations in rotations abound the Z-axis. After the first rotation around the Z-axis a deviation of $F_{x,R_z,90^\circ} \approx -0.25N$, $F_{y,R_z,90^\circ} \approx 0.55N$ and $F_{z,R_z,90^\circ} \approx -0.15N$ is observed. After the second rotation a deviation of $F_{x,R_z,180^\circ} \approx -0.75N$, $F_{y,R_z,180^\circ} \approx 0.22N$ and $F_{z,R_z,180^\circ} \approx -0.32N$ is observed. Observe that the $F_{x,R_z,180^\circ} \approx -0.75N$ deviation after the second rotation is larger in magnitude the the estimated equipment weight force $F_{equipment} = 0.5886N$ calculated in section 5.2.2.

Several sources could explain the deviation observed with rotations around the Z-axis. Firstly, rotations around the Z-axis cause movements in the leather wristband. The wristband weight of 7 grams could be recorded by the sensitive F/T sensor, but it would only explain very small parts of the deviations. Rotations around the Z-axis is also sensitive to inaccuracies in the center of mass estimate presented in figure 3.8 that could explain small inaccuracies. This is because the calculated distance from the sensor calibration point to the center of mass is defined on the F/T sensors Z-axis. The observed difference $|F_{x,R_z,180^\circ}| > F_{equipment}$ in the second rotation in the Z-axis support this possibility. The most probable source is the inability to properly compensate for the implementation challenges with the F/T sensor mounting bias. The discussion behind the preventive security measure resulting in a double software bias is presented in section 5.2.2.

The dead-band filter with a force interval of $\pm 1N$ prevents unwanted drifting movements from the observed deviations $< 1N$, but this solution can become problematic with a heavier end-effector design. The overall performance assessment is that the gravity compensation works as intended but due to preventive security measures it is not able to compensate properly for all the masked gravity components of the equipment weight. The gravity compensation combined with the dead-band filter is good enough for practical implementations, but the observed deviations in rotations around the Z-axis are unsatisfactory and the dead-band solution does not scale well with an increase in equipment weight.

6.2 Executive controller design

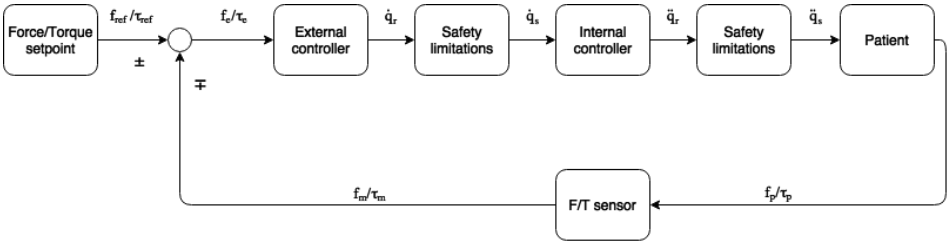


Figure 6.2: Executive controller design for the robotic rehabilitation system. The interactions between the external controller and the internal firmware controller is highlighted.

The lack of access to low-level joint torque control poses limitations upon the external controller design. The lowest control level available to the external controller is a joint velocity interface $\dot{q}_r$. The security measures described in table 5.3 is imposed upon the joint velocities $\dot{q}_s$ before they are sent to the internal firmware controller. Unfortunately, the details regarding the internal controller are unknown as the producer regards this information as classified trade secrets. This poses another significant limitation since the internal controller essentially works as a ”black box” during the development of the external controller. The only reasonable assumption that is made is that the internal controller reaches the desired joint velocities by changing the joint accelerations $\ddot{q}_r$ through manipulating

the individual joint motor voltages. The safety-related security functions described in table 5.2 is imposed upon the joint accelerations $\ddot{q}_s$ before they are executed by the robotic manipulator. Figure 6.2 shows an illustration of the executive control loop for the robotic rehabilitation system.

6.3 Low-level control strategy

The low-level control strategy is named robotic rehabilitation compliance mode as the only function is to follow any movements generated by the patient in all six degrees of freedom. The force- and torque reference value f_{ref}/τ_{ref} is set to zero. Patient-generated forces and torques f_p/τ_p are measured in the tool frame by the F/T sensor f_m/τ_m, compared to the reference values before the force/torque errors f_e/τ_e are passed to the external controller. Compliance mode is implemented directly in the tool frame to encourage the natural interactions between the end-effector and stroke patients.

Incremental controller design

The mathematics in the internal controller is unknown. It is therefore unclear how the system will react to different implementations of an external controller. In the simplest and best case scenario, the internal controller contains a proportional part and derivative part that is imposed upon the velocity control signal before executing the calculated acceleration control signal. This would result in all force/torque control inputs being integrated once and kinematically transformed inside the external controller before being derived once inside the internal controller and subsequently executed. Undoubtedly, this process will greatly amplify any signal noise. In a worst case scenario, unknown dynamics of the internal controller could result in control signal conflicts between the external and internal controller were they start to counteracts each others control signals. The systems behaviour and outcome of a fight between the controllers would be highly unpredictable.

An incremental controller design process is suggested to investigate the system dynamics. The goal is providing the best possible solution for indirect force and torque control with the limitations imposed by the internal controller. The incremental process involves first implementing and tuning a P-controller, PI-controller, PD-controller and finally a PID-controller as the external controller. Normally, joint position q_i regulation in robotic manipulator only require a P- or PI-controller. In this application, the external controller seeks to regulate acceleration $\ddot{q}_i$ indirectly through a joint velocity $\dot{q}_i$ control interface. This application could benefit from a derivative part, but it will make the system more susceptible to noise and random values.

Re-designed external controller

The external controller would originally handle force- and torque inputs from the stroke patient with a single set of gain parameters. Experimental tuning with this controller design proved to be difficult, as achieving an antiquate response to transitional forces resulted in an over-aggressive response to rotational torques and vice versa. Several attempts at a

compromise did not yield the desired behaviour. Generating two different responses to the control inputs required a re-designed of the external controller to incorporate two sets of gain parameters, one set for transitional forces and one set for rotational torques.

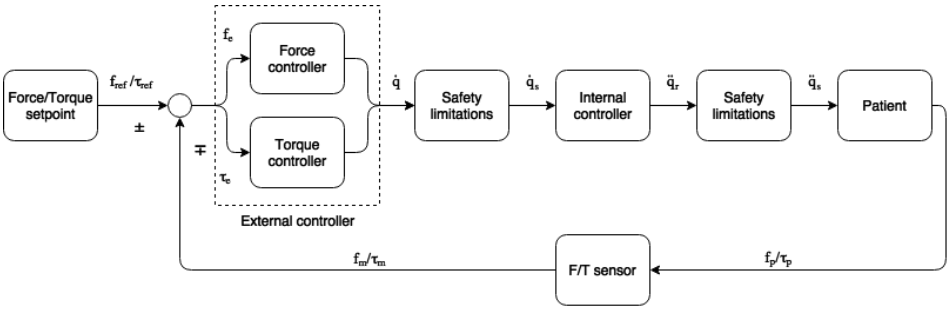


Figure 6.3: Re-design of the executive controller design for the robotic rehabilitation system.

Pseudo code controller implementation

The entire code repository for the robotic rehabilitation system is available on GitHub[1] [13] or by contacting the Department of Engineering Cybernetics at NTNU. The incremental controller design process is realized by a discrete controller implemented in the programming language C++ on the external computer. Pseudo code illustrate the external controller design with a PID-controller implementation and should result in a preferred gain parameter scale where $K_p > K_i > K_d$.

```
while(running_safely){
  //Force controller in 3 DOF
  F_measured = gravity_compensation(Q_position_current, F_patient);
  F_error = F_reference - F_measured;
  F_integrator = F_integrator + F_error;
  F_derivative = F_error - F_previous_error;
  u_F = Kp*F_error + Ki*F_integrator + Kd*F_derivative;

  //Torque controller in 3 DOF
  T_measured = T_patient;
  T_error = T_reference - T_measured;
  T_integrator = T_integrator + T_error;
  T_derivative = T_error - T_previous_error;
  u_T = Kp_T*T_error + Ki_T*T_integrator + Kd_T*T_derivative;

  vw[6] = [u_Fx, u_Fy, u_Fz, u_Tx, u_Ty, u_Tz];
  Q_velocity_desired = inverse_Jacobian(Q_position_current, vw);
  if(Q_velocity_desired != safe){
    running_safely = false;
    break;
  }
  UR5.send_speed_command(Q_velocity_desired);
}
```

[1] https://github.com/madlaa/NTNU-master-thesis

The controller thread runs at $125Hz$ and receives raw force/torque data through a separate thread that runs slightly faster the sampling frequency of $250Hz$. The raw torque data is processed directly while the raw force data is gravity compensated before being passed into a control signal error. The errors are integrated and derived for each control iteration. The accumulated control values are multiplied with the relevant controller gains and added to a task space vector.

vw[6] = [u_Fx, u_Fy, u_Fz, u_Tx, u_Ty, u_Tz];

The task space vector consists of three linear velocities and three angular velocities for the end-effector. The desired velocity control inputs are generated from the task space vector with an inverse kinematic solver. Several safety checks are performed before the control inputs are sent to the internal firmware controller in the form of a URScript speedj command.

Alternative implementation

Other code implementations of the integrator and derivative parts have been considered for both forces and torques,

F/T_integrator = F/T_integrator + (F/T_error)*iterationTime;
F/T_derivator = (F/T_error - F/T_previous_error)/iterationTime;

This implementation is however not preferred as it negatively effects the system dynamics. With a maximum hardware limited controller frequency of $125Hz \approx 0.008s$, the small iteration time results in massively scaled contributions from the derivative part for even the smallest values of K_d.

Controller tuning

Normal methods of tuning a controller involve system modeling or utilizing tuning parameters derived from the amplitude and frequency of the maximum proportional gain. However, the application in this master thesis involves several nonlinearities and uncertainties connected with the executive controller loop. The nonlinearities connected with human movement, especially prevalent in stroke patients, is highly complex and difficult to model. In addition, both the mounting pedestal and table is not bolted securely to the ground, allowing for some small nonlinear movements under heavy load. Modeling these nonlinearities would be time-consuming and modeling human movement would most likely produce insufficiently accurate results for practical use. The unknown outcome of a potential fight between the integral and external controller is also a major factor. The simplified and practical solution is to experimentally tune the two sets of controller gains parameters for the different iterations of the external controller.

Controller testing setup

Two simple laboratory setups are needed to experimentally tune the two sets of gain parameters for transitional forces and rotational torques. All tests are performed with Prototype 1.1. The setup for tuning the force controller consists of positioning the end-effector

$1cm$ above a cardboard box placed on the table with the Z-axis of the end-effector parallel with the vertical plane. The force controller is then given a vertical reference force of $2N$ pointed downwards for a duration of $5s$. The setup for tuning the torque controller consists of positioning the end-effector $1cm$ above the cardboard box with the Z-axis of the end-effector parallel with the horizontal plane. The torque controller is then given a rotational reference torque of $1Nm$ around the Z-axis for a duration of $5s$. The low-pass filtration with a $5Hz$ cut-off frequency is enabled, but some of the safety features in the external controller are disabled as it might interfere with the results. This includes the virtual workspace and the emergency shutdown procedures. The cardboard box is a preventive safety measure that is intended to structurally fail before significant damage to personnel or hardware can occur. The cardboard box transfer forces and torques into the table fairly consistently, but introduces some potential nonlinearities due to material strength and elasticity. The assessment is that this effect is insignificant for the small forces and torque utilized in the experimental tuning as long as the structural integrity remains intact.

After each experimental tuning, the results are compared with a subjective assessment of the controller behaviour for each gain parameter. The subjective assessment is based on the safety checklist presented in chapter 3.6.1 and appendix C, consisting of incremental interactions between the operator and the manipulator in all six degrees of freedom.

6.3.1 Force control

Proportional gain estimation

The force P-controller is tested with numerous proportional gain parameters. The results from four proportional gain parameters $K_p = 0.0025$, $K_p = 0.0050$, $K_p = 0.0075$ and $K_p = 0.0100$ are highlighted and illustrated in figure 6.4.

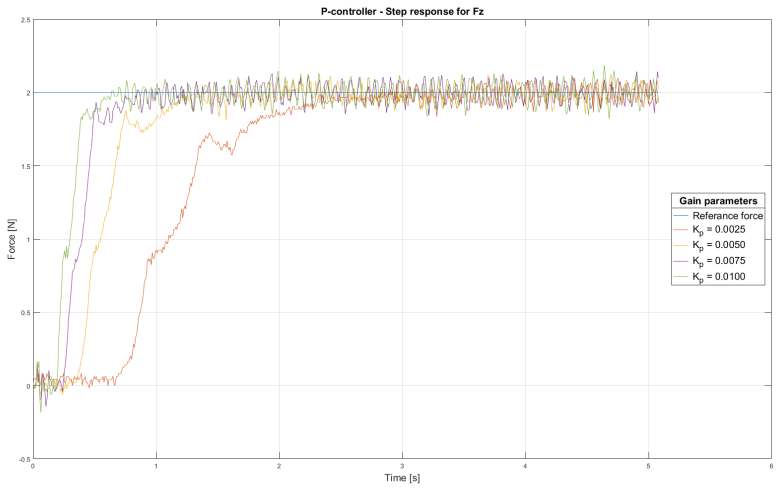


Figure 6.4: Experimental testing of the force P-controller with four different proportional gain parameters.

Performance analysis

The results show that the lowest gain parameter $K_p = 0.0025$ is not able to close a gap of $1cm$ within a $3s$ time period, which is a rise time that is unacceptably slow. The two middle gain parameters $K_p = 0.0050$ and $K_p = 0.0075$ reach the reference value within $1s$ and $1.5s$ respectively. The highest gain parameter $K_p = 0.0100$ reaches the reference value within $0.8s$. The experimental results indicate that a proportional gain parameter between $K_p = 0.0050$ and $K_p = 0.0100$ provide an excellent step response. The step response appears to be critically damped without overshoot and an insignificant steady-state error from the reference value. Signal noise around the reference value is observed for all four proportional gain parameters.

The three best gain parameters are experimentally tested with the subjective assessment method. The cautiously low proportional gain parameter of $K_p = 0.0050$ is subjectively preferred as it provides the best compromise between a responsive reaction and a safe behaviour with minimized unwanted oscillations.

Proportional-Integral gain estimation

The force PI-controller is tested with the preferred proportional gain of $K_p = 0.005$ and numerous integral gain parameters. The results from four integral gain parameters $K_i = 0.00010$, $K_i = 0.00015$, $K_i = 0.00020$ and $K_i = 0.00025$ are highlighted and illustrated in figure 6.5.

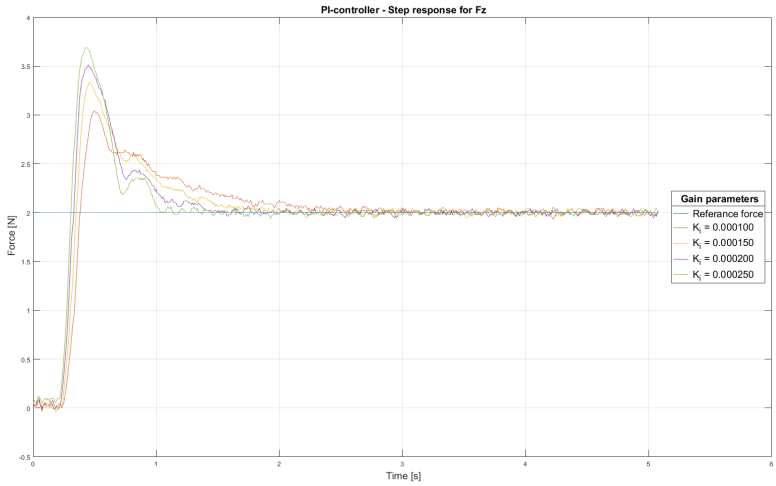


Figure 6.5: Experimental testing of the force PI-controller with $K_p = 0.005$ and four different integral gain parameters.

The step response from all four tests show some highly interesting results. The PI-controller improves the rise time as all four integral gain parameter reach the reference value within $0.5s$, but it introduces a significant overshoot in the step reasons. The overshoot increases incrementally relative with the increase in the integral gain parameters. For the largest value $K_i = 0.00025$ the overshoot exceeds $3,68N$, almost a doubling of the reference force of $2N$. The overshoot behaviour is easily observable in the experimental test setup and the signal noise around the reference value appear to be decreased for all the integral gain parameters highlighted in figure 6.5.

The subjective assessment reveals a behaviour that is more aggressive and responsive, and the overshoot can easily be felt by the operator for all the integral gain parameters. Moving the manipulator is perceived as moving a large mass with a significant inertia. This could pose a potential danger to the stroke patients and should be avoided in the final force controller implementation.

Proportional-Derivative gain estimation

The force PD-controller is tested with the preferred proportional gain of $K_p = 0.005$ and numerous derivative gain parameters. The results from four derivative gain parameters

$K_d = 0.00075$, $K_d = 0.00750$, $K_d = 0.02500$ and $K_d = 0.07500$ are highlighted and illustrated in figure 6.6.

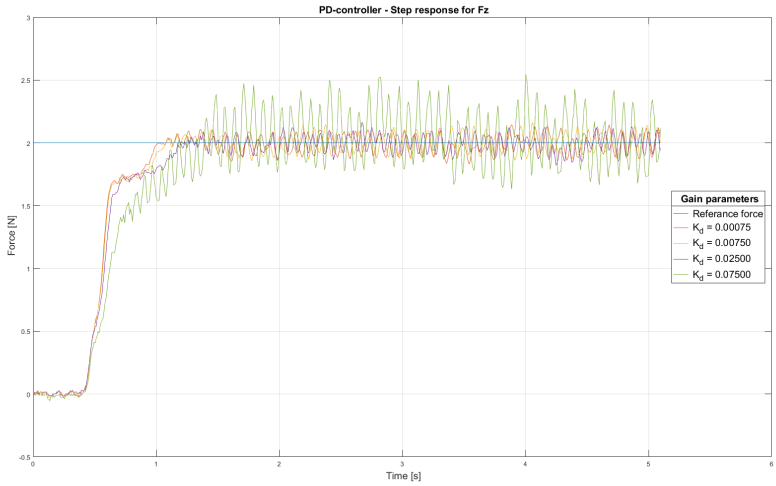


Figure 6.6: Experimental testing of the force PD-controller with $K_p = 0.005$ and four different derivative gain parameters.

The results in figure 6.6 show that the PD-controller generally introduces more signal noise to the step response for the larger derivative gain parameters. The rise time and signal noise increase incrementally relative with the increase in the derivative gain parameters. The highest derivative gain parameter $K_i = 0.07500$ introduces signal noise that oscillates with a magnitude of $\approx \pm 0.5N$ around the reference value. The noise dominates the results, but the four test show no tendencies for a steady-state error from the reference value can be observed.

The subjective assessment reveals an insignificant change in behaviour for small derivative gain parameters and a behaviour that is twitching and nervous for larger derivative gain parameters. The noise and twitching produced by the larger derivative gain parameters could pose a potential danger to the stroke patients and should be avoided in the final force controller implementation.

Proportional-Integral-Derivative gain estimation

The force PID-controller is tested with a preferred proportional gain of $K_p = 0.005$, the smallest integral gain tested $K_i = 0.000075$ and numerous derivative gain parameters as they had the smallest negative impact on the system behaviour. The results from four derivative gain parameters $K_d = 0.01$, $K_d = 0.05$, $K_d = 0.10$ and $K_d = 0.25$ are highlighted and illustrated in figure 6.7.

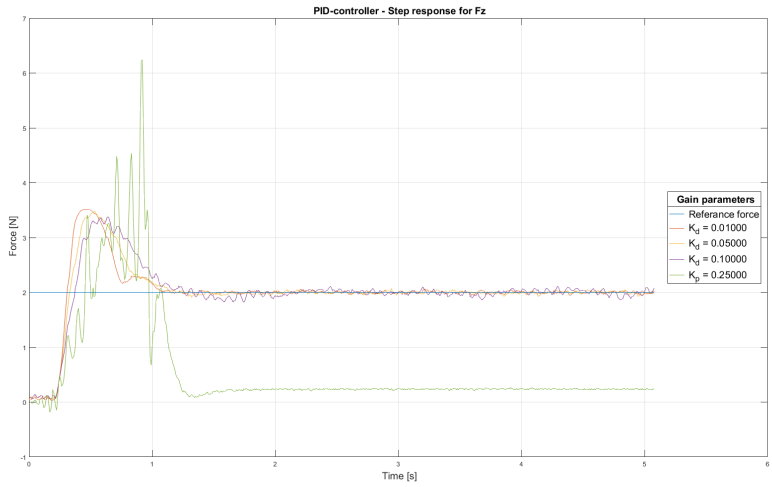


Figure 6.7: Experimental testing of the force PID-controller with four different gain parameters.

The results show that the three smallest derivative gain parameters, with a 50% incremental value increase, are not able to counteract the overshoot. The results from the force PD-controller revealed an increasing signal noise amplification for larger derivative gain parameters. Despite this, it is attempted to see if it is even possible to reduce the overshoot caused by the integral part. The final derivative gain parameter is increased by 150% to $K_d = 0.25$. This produced an unstable step response in the manipulator that partly crushed the cardboard box before the operator triggered the emergency shutdown button on the teach pendant. Obviously, this behaviour is unacceptable for the final force controller implementation.

Careful subjective assessment on the three lower derivative gain parameters reveal a twitching behaviour that is prone to unwanted oscillations while still acting as a large mass with significant inertia. This behaviour poses a danger to the stroke patients and should be avoided in the final force controller implementation.

6.3.2 Torque control

The laboratory setup is recreated after the final experimental tuning of the force PID-controller partly crushed the cardboard box. The new cardboard box is identical and is therefore subject to the same laboratory setup dimensions and material assessments as the original.

Proportional gain estimation

The torque P-controller is tested with numerous proportional gain parameters. The results from four proportional gain parameters $K_{p,T} = 0.1$, $K_{p,T} = 0.25$, $K_{p,T} = 0.5$ and

$K_{p,T} = 0.75$ are highlighted and illustrated in figure 6.8.

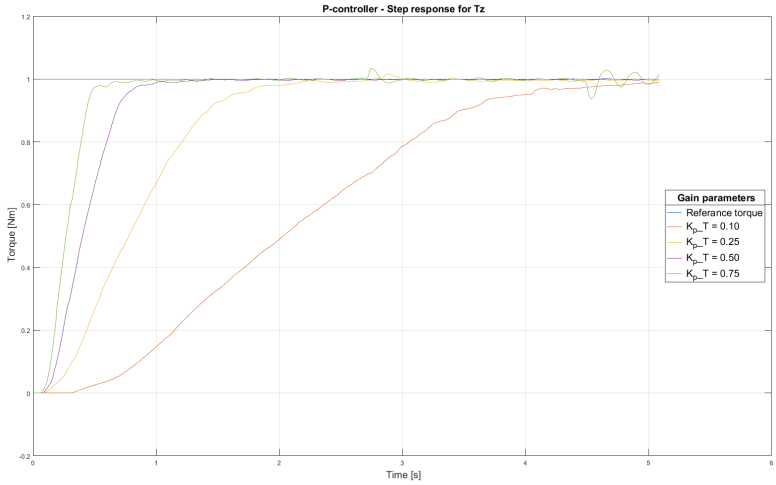


Figure 6.8: Experimental testing of the torque P-controller with four different proportional gain parameters.

The results show that the lowest gain parameter $K_{p,T} = 0.1$ is not able to close a rotational gap of approximately $\approx 1.1°$ within the entire $5s$ time period, this is a rise time that is unacceptably slow. The two middle gain parameters $K_{p,T} = 0.25$ and $K_{p,T} = 0.5$ reach the reference value within $1s$ and $2s$ respectively. The highest gain parameter $K_{p,T} = 0.75$ reaches the reference value within $0.6s$. The experimental results indicate that a proportional gain parameter between $K_{p,T} = 0.25$ and $K_{p,T} = 0.75$ provide an excellent step response. As seen previously with the force controller, the step response appears to be critically damped with an insignificant steady-state error from the reference value.

The three best gain parameters are then experimentally tested with the subjective assessment. The cautiously low proportional gain parameter of $K_{p,T} = 0.4$ is preferred as it provides the best compromise between a responsive reaction and a safe behaviour with minimized unwanted oscillations.

Proportional-Integral gain estimation

The torque PI-controller is tested with the preferred proportional gain of $K_{p,T} = 0.4$ and numerous integral gain parameters. The results from four different integral gain parameters $K_{i,T} = 0.00025$, $K_{i,T} = 0.00050$, $K_{i,T} = 0.00075$ and $K_{i,T} = 0.00100$ are highlighted and illustrated in figure 6.9.

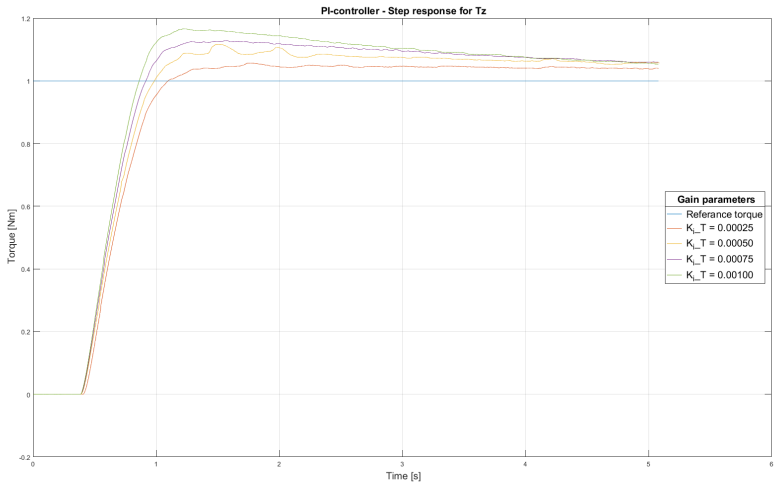


Figure 6.9: Experimental testing of the torque PI-controller with four different gain parameters.

The torque PI-controller improves the rise time for all four integral gain parameter. The three highest values reach the reference value within a time period of $[0.85s - 1s]$ and the smallest value reach the reference value after $1.1s$. The torque PI-controller introduces a small overshoot with a steady state error in the step response, noticeably differentiating from the dominating overshoot dynamic without a steady state error observed in the force PI-controller. The steady state error and overshoot increases incrementally relative with the increase in the integral gain parameter, for the gain parameters $K_{i,T} = 0.00100$ and $K_{i,T} = 0.00050$ the overshoot peaks at $1.165Nm$ and $1.115Nm$ respectively and both converge towards a steady state error of $1.06Nm$.

The subjective assessment reveals that the PI-controller is more susceptible to unwanted oscillations, especially with movements that activate the force P-controller and torque PI-controller simultaneously. The unwanted behaviour is not directly danger to the stroke patients, but could easily be perceived as uncomfortable or unnerving. Insecurity in rotation movements is perceived as particularly uncomfortable compared to translation movements.

Proportional-Derivative gain estimation

The torque PD-controller is tested with the preferred proportional gain of $K_{p,T} = 0.4$ and numerous integral gain parameters. The results from four different integral gain parameters $K_{d,T} = 0.00050$, $K_{d,T} = 0.05000$, $K_{d,T} = 5.00000$ and $K_{d,T} = 20.00000$ are highlighted and illustrated in figure 6.10.

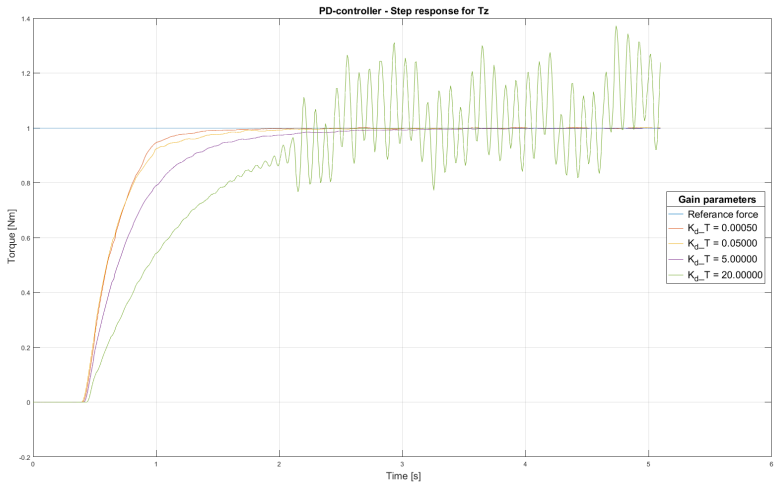


Figure 6.10: Experimental testing of the torque PD-controller with four different gain parameters.

The torque PD-controller decrease the rise time for all four derivative gain parameter. The three lowest values converge on the reference value within an approximate time period of $[2s - 4s]$ without a steady state error. The rise time and convergence decrease incrementally relative with the increase in the derivative gain parameter. The noise amplification behaviour is more subtle and is only seen for very large derivative gain parameter illustrated by the largest derivative gain parameter $K_{d,T} = 20.00000$. The largest derivative gain parameter $K_{d,T} = 20.00000$ amplified the noise and generated oscillatory rotational movements on the surface of the cardboard box with a high frequency. The rotational movements do no effect the structural integrity of the cardboard box due to the small magnitudes involved.

The subjective assessment of the three smaller derivative gain parameter reveals that the PD-controller is almost indistinguishable from the torque P-controller with a small derivative gain parameter. The increase in noise amplification is only felt for the two middle derivative gain parameter. The PD-controller is susceptible to twitching, especially with movements that activate the force P-controller and torque PD-controller simultaneously. This is particularly observed during rapid movements. The unwanted behaviour is not directly danger to the stroke patients, but could easily be perceived as uncomfortable or unnerving. Insecurity in rotation movements is perceived as particularly uncomfortable compared to translation movements.

Proportional-Integral-Derivative gain estimation

The torque PID-controller is tested with a proportional gain of $K_{p,T} = 0.4$, a lower middle integral gain of $K_{i,T} = 0.00050$ and numerous derivative gain parameters as they had the smallest negative impact on the system behaviour. The results from four different

derivative gain parameters $K_{d,T} = 1.0$, $K_{d,T} = 2.5$, $K_{d,T} = 5.0$ and $K_{d,T} = 7.5$ are highlighted and illustrated in figure 6.11.

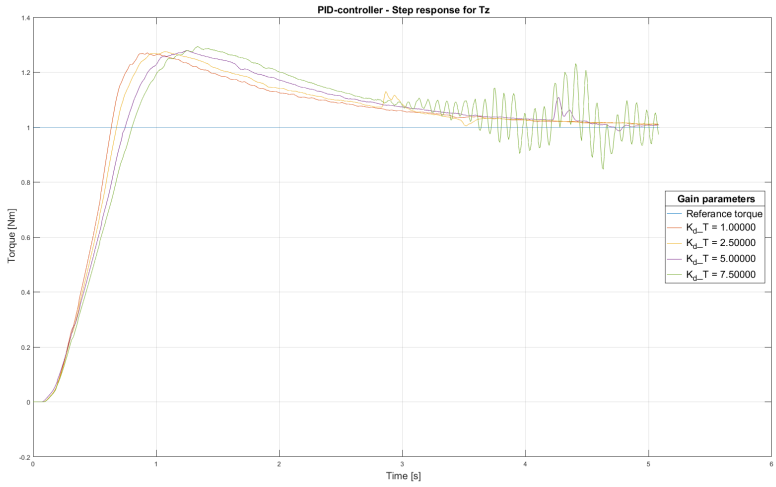


Figure 6.11: Experimental testing of the torque PID-controller with four different gain parameters.

The results show that all the derivative gain parameters slightly increase the overshoot peak value to $> 1.27Nm$, but the steady state error seems to be eliminated as the three smallest values shown a tendency towards slowly converging on the reference torque. The largest gain parameter $K_{d,T} = 7.5$ shows a significant and observable instability towards the end of the test period. Careful subjective assessment on the three lower derivative gain parameters reveals a similar behaviour, best described as aggressive and prone to unwanted oscillations.

6.3.3 Summary

The incremental controller design process produced several interesting results about the internal and external controller dynamic. The lack of a steady state error for both P-controllers is especially interesting. This behaviour is caused by a known phenomenon where the manipulator is acting as a natural integrator to the velocity control inputs. Implementing a P-controller on a velocity $\dot{q}_i$ results in indirectly implementing a PI-controller on the position q_i. Explaining why both P-controller display a step response seems to be critically damped is more challenging, but it might be caused by a good design and perfect tuning of the internal controller. Another interpretation is that the external controller implemented with two P-controllers consistently loses a control signal fight with the internal controller and therefore result in the desired behaviour.

Both the PI-controllers increased the rise time but they acted differently upon the step response. The force PI-controller introduced a significant overshoot, while the torque PI-

controller introduced a slight overshoot and a steady state error.

Both PD-controllers decreased the rise time but they acted differently upon the step response. The force PD-controller quickly introduced a twitching behaviour prone to unwanted oscillations. The torque controller also displayed a twitching behaviour, but it was more restrained and only observable for large derivative gain parameters. An important subjective observation is that aggressive rotational behaviour is perceived as significantly more uncomfortable to the operator as an aggressive transitional behaviour.

Both the force and torque PID-controllers did not counteract for the overshoot and generally resulted in an aggressive behaviour prone to unwanted oscillations that are unwanted in a final implementation. The torque PID-controller shows a tendency towards eliminating the steady-state error by slowly converging upon the reference value.

The experimental verification results from both P-controllers are exceptionally good. All other controller designs show worse performance in both the experimental verification tests and the subjective assessments. The suggestion is therefore to design the low-level control strategy around a force P-controller and a torque P-controller. This design provides the best results in the experimental tuning and a safe behaviour in the subjective assessment. The good step response, resilience towards unwanted oscillations and twitching together with the increase in overall security is paramount for the robotic rehabilitation system. The torque- and force P-controllers is implemented with the preferred proportional gain parameter of $K_p = 0.005$ and $K_{p,T} = 0.4$ respectively.

6.4 High-level control strategies

The high-level control strategies seek to implement active rehabilitation strategies were the stroke patient is required to constantly participate in all movements or tasks. Several design choices are affected by the assignment interpretation and the functional approach towards realizing the goals and objectives in this thesis.

Path planning

An important design choice regards implementations that rely on a predefined path that the patient must follow. Utilizing predefined paths could require the user to calibrate the robotic rehabilitation system with the physical dimensions, complications and capabilities for each individual stroke patient. These variables pose a significant challenge to a high-level control strategy with a predefined path. For example, a large male patient and a small female patient would follow very different paths while attempting to perform a complex task like grasping and drinking from a cup placed on the table. A path that feels natural for one of the patients might feel unnatural or infeasible to the other patient and vice versa. In addition, high-level control strategies that require the developer to individually implement exercises into the code should be avoided. Implementations with a more dynamic and modular approach are preferred at this early stage of development. Determining the best path to follow is at this stage viewed as discretionary assessments that should be per-

formed by professional physical therapists working together with the individual patient. This is in accordance with the intended use of the robotic rehabilitation system as a helpful rehabilitation tool used concurrently together with traditional stroke rehabilitation. For the application in this master thesis detailed path planning is currently seen as generally limiting and is therefore avoided in the development of the robotic rehabilitation system.

Robotic rehabilitation modes

Four different high-level robotic rehabilitation control strategies are designed and developed based on the low-level control strategy. Realizing the different strategies is mainly achieved by manipulating the force-/torque reference values f_{ref}/τ_{ref}. The developed high-level control strategies are illustrated in figure 6.12.

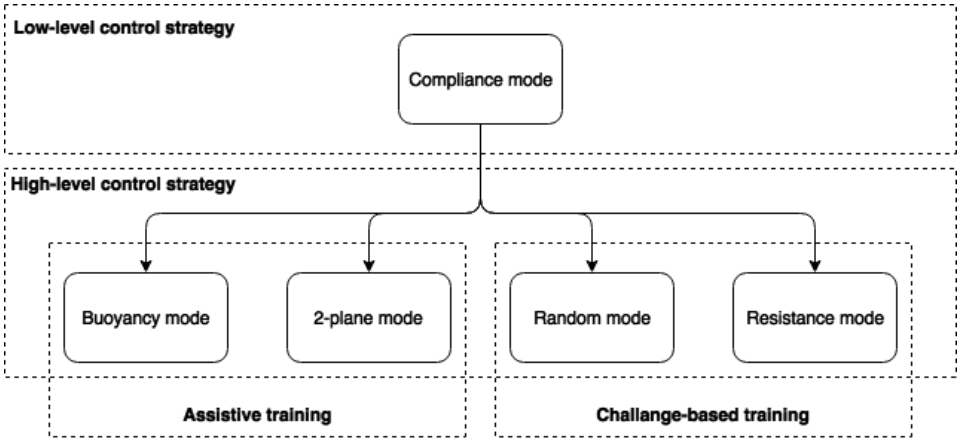


Figure 6.12: High-level control strategies for the robotic rehabilitation system.

The four different high-level robotic rehabilitation control strategies can be divided into two subgroups, assistive therapy and challenge-based therapy. Assistive therapy seeks to aid the patients in performing a movement or task while challenge-based therapy seeks to interfere or hinder the patient in performing a movement or task. Both strategies seek to improve the patients hand-eye coordination, muscle strength and regaining upper-limb functionality. The two assistive robotic rehabilitation strategies are named 2-plane mode and buoyancy mode, and the two challenge-based robotic rehabilitation strategies are named resistance mode and random mode.

2-plane mode

The recommendation in section 2.1.2 cautioned about implementing constraint-induced movement therapy, mainly based on the lack of scientific documentation on the positive effect of constraint-induced movement therapy on stroke patients. The robotic rehabilitation 2-plane mode is designed to challenge the established view and provide a basis for further scientific documentation on constrained induced assistive movement therapy on

stroke patients. The 2-plane mode is aimed at the weakest patients as it contains all patient generated movements to a predefined two-dimensional plane. The mode is classified as assistive therapy since it is intended to aid the weakest patients that struggle with basic muscle coordination and that lack the muscle strength needed to lift their own arm weight. Constraint-induced assistive movement therapy would be a more precise classification of the robotic rehabilitation 2-plane mode. The 2-plane mode is implemented directly into the tool frame. This enables the user to quickly and easily change the initial position the end-effector into a pose that enables rehabilitation exercises in the desired two-dimensional plane of motion. This enables the patient to focus entirely on the movement and muscle coordination in a desired plane of motion.

Buoyancy mode

The robotic rehabilitation buoyancy mode seeks to aid the patient in performing movement or tasks in all six degrees of freedom by reducing the perceived weight of the stroke patient arm. The buoyancy mode enables the user to define the amount of force that the manipulator constantly applies to counteract the weight of the stroke patient arm. The buoyancy vector is defined by the user in the world frame and kinematically transformed before it is sent to the controller. The maximum allowed compensation force of $25N$ is calibrated based on a healthy male with average height and weight perception of when the manipulator is taking most of the weight of the arm while still feeling comfortable with the interaction.

Resistance mode

The robotic rehabilitation resistance mode constantly applies a resistance to movements generated by the stroke patient in all six degrees of freedom. It is aimed at stroke patients that master basic hand-eye coordination and seeks to improve their muscle strength. Resistance mode is not classified as constrain induced movement therapy since it is resisting, not constraining movements. The mode enables the user to adjust the level of resistance in all six degrees of freedom according to the capabilities of the stroke patient. The resistance mode is implemented directly in the tool frame on the force- and torque control inputs. The maximal resistance is calibrated based on a healthy male with average height and weight perception of when the manipulator negates almost all patient generated movements in the associated degree of freedom.

Random mode

The robotic rehabilitation random mode provides a real challenge to all stroke patients with basic upper-limb functionality and can be utilized in moderate to later stages in the rehabilitation process. The effect is achieved by constantly generating vectors with a random magnitude between $\pm 15N$ and a random direction. The random vector is applied as a new force reference f_{ref} and maintained for a random time interval between $[2s-4s]$. The newly generated random vectors are implemented directly in the tool frame and are applied slowly over time to the force reference to avoid rapid changes in the end-effector velocity. The gentle step-up in magnitude over time prevents rapid velocity movements that could

trigger a spastic muscle reaction or be perceived as uncomfortable by the patients. Random mode enables the user to choose the desired magnitude in all three transitional degrees of freedom based on the capabilities of the stroke patient. Early testing and development revealed that several subjects perceived random rotational disturbances as uncomfortable. Even though random rotational disturbances are not directly dangerous to the patients due to the safety functionality it is disabled as a precautionary security measure. Performing a movement or task is made significantly more difficult while fighting randomly generated movements by the robot manipulator. The maximal magnitude of the random movements is calibrated from what a healthy male with average height and weight finds moderately challenging while performing a delicate movement or task. Movements and tasks close to the face or other delicate parts of the body should not be performed in this rehabilitation mode as a precautionary security measure.

Chapter 7

Experimental verification

The finished external controller design needs to be experimentally verified in the laboratory setup. This includes the external safety mechanisms and four high-level control strategies designed for the robotic rehabilitation system.

7.1 Emergency shutdown procedures

The external controller initiates an emergency shutdown procedure if it detects forces that exceed $50N$ or torques that exceed $8Nm$ in any of the degrees of freedom. Six individual safety tests, one for each degree of freedom, is conducted and recorded. In each test, the operator seeks to apply a force/torque with a linear growth in magnitude until the external controller throws an exception and begins the emergency shutdown procedure. The differences in growth rate and rise-time are explained by the inconsistencies in the operator's movements, but it is not relevant for this safety test. The emergency shutdown procedure is conducted in compliance mode with all joints locked in the initial angular position to prevent manipulator movement from affecting the outcome. The results are presented in fig 7.1.

Performance analysis

The six test results illustrated in figure 7.1 show an approximately linear growth rate in the applied forces/torques with a different rise-time to the limit value. The three transitional force tests show an emergency shutdown procedure being triggered when $F_x, F_y, F_z > 50N$ at time $2.9040s, 3.6560s$ and $2.5760s$ respectively. The three rotational torque tests show an emergency shutdown procedure being triggered when $T_x, T_y, T_z > 8Nm$ at time $4.1840s, 5.8080s$ and $3.1520s$ respectively.

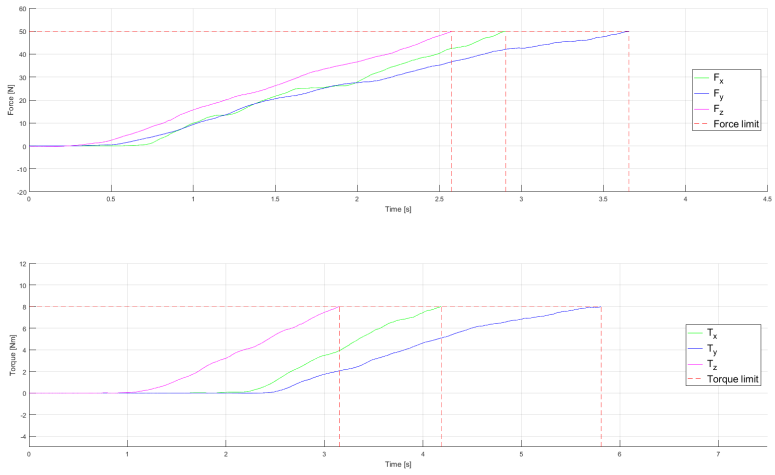


Figure 7.1: Six individual emergency shutdown tests, one for each degree of freedom. The operator applies a force/torque with an approximately linear growth until the external controller initiates an emergency shutdown procedure.

7.1.1 Virtual workspace

The virtual workspace for the UR5 manipulator presented in chapter 5 is experimentally verified in the robotic rehabilitation compliance mode. Four verification tests are suggested, an individual test for the outer-, inner-, lower- and upper soft boundaries that together constitute the virtual workspace. The TCP trajectory for each test is mapped in Cartesian space with a center-point in the world frame. The force readings are recorded in the tool frame and kinematically transformed into the world frame before being presented in the results. Each test seeks to penetrate the relevant soft boundary three times at different locations during a $30s$ test period. The operator seeks to maintain a constant position outside the virtual workspace with a constant force for a short time period.

The data is presented with some visual aid methods that seek to improve the visual interpretation of the results. The soft boundaries in the virtual workspace is highlighted in a gradient color relative to the current height, purple at the lower soft boundary and yellow at the upper soft boundary. The force readings turn red when the TCP is outside the soft boundaries, and the TCP trajectory highlights the relevant radius- or height correction vectors applied by the external controller to guide the TCP back inside the virtual workspace. The virtual workspace verification test results are presented in figure 7.2, 7.3, 7.4 and 7.5.

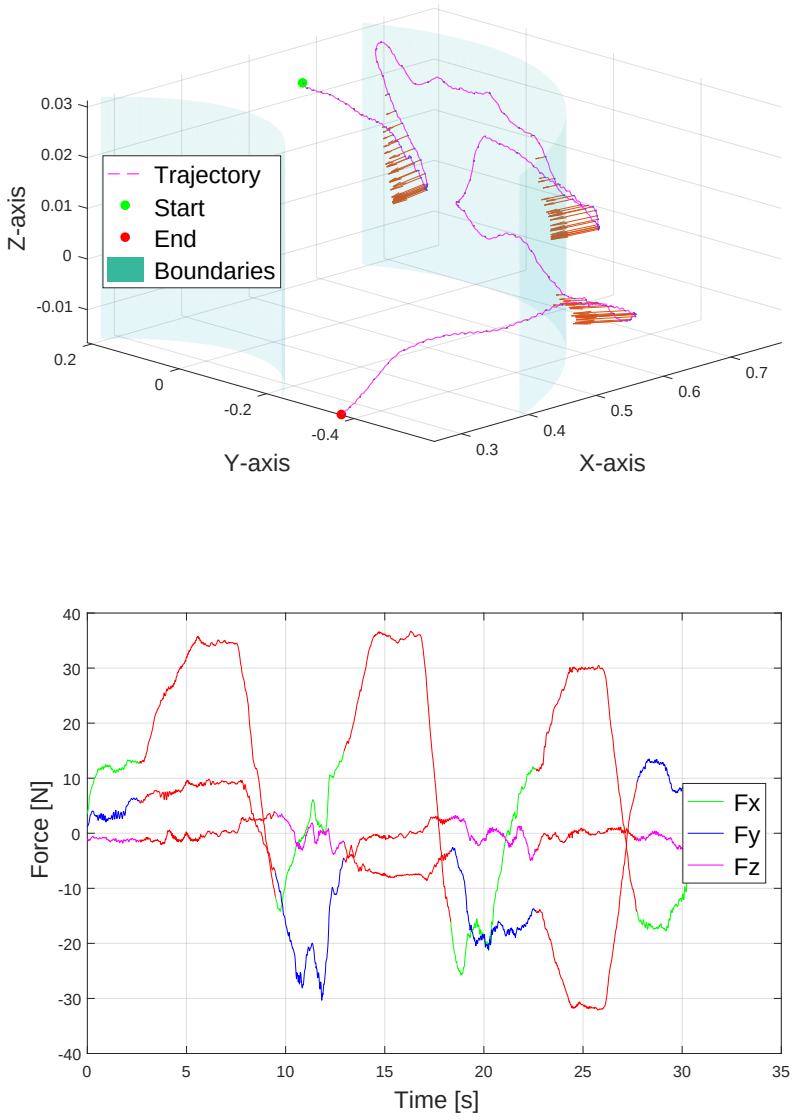


Figure 7.2: Experimental verification test of the outer cylindrical soft boundary in the virtual workspace. Force readings turn red when the TCP is outside the virtual workspace.

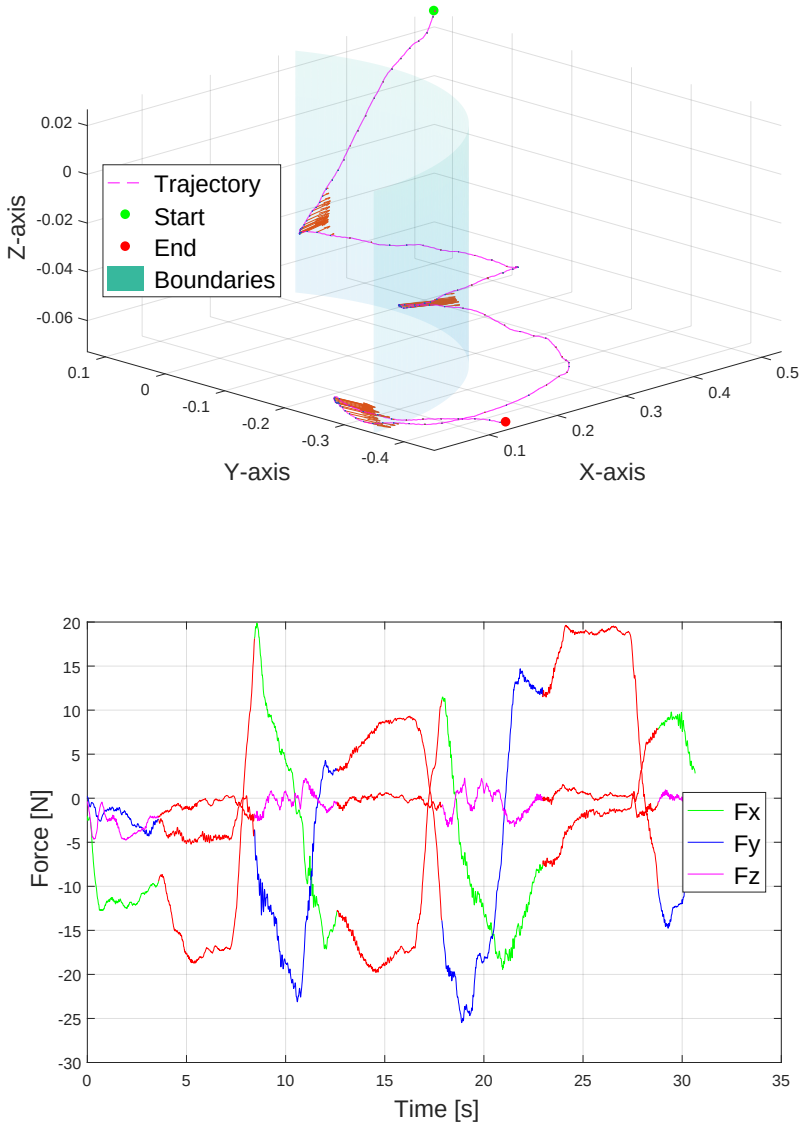


Figure 7.3: Experimental verification test of the inner cylindrical soft boundary in the virtual workspace. Force readings turn red when the manipulator is outside the virtual workspace.

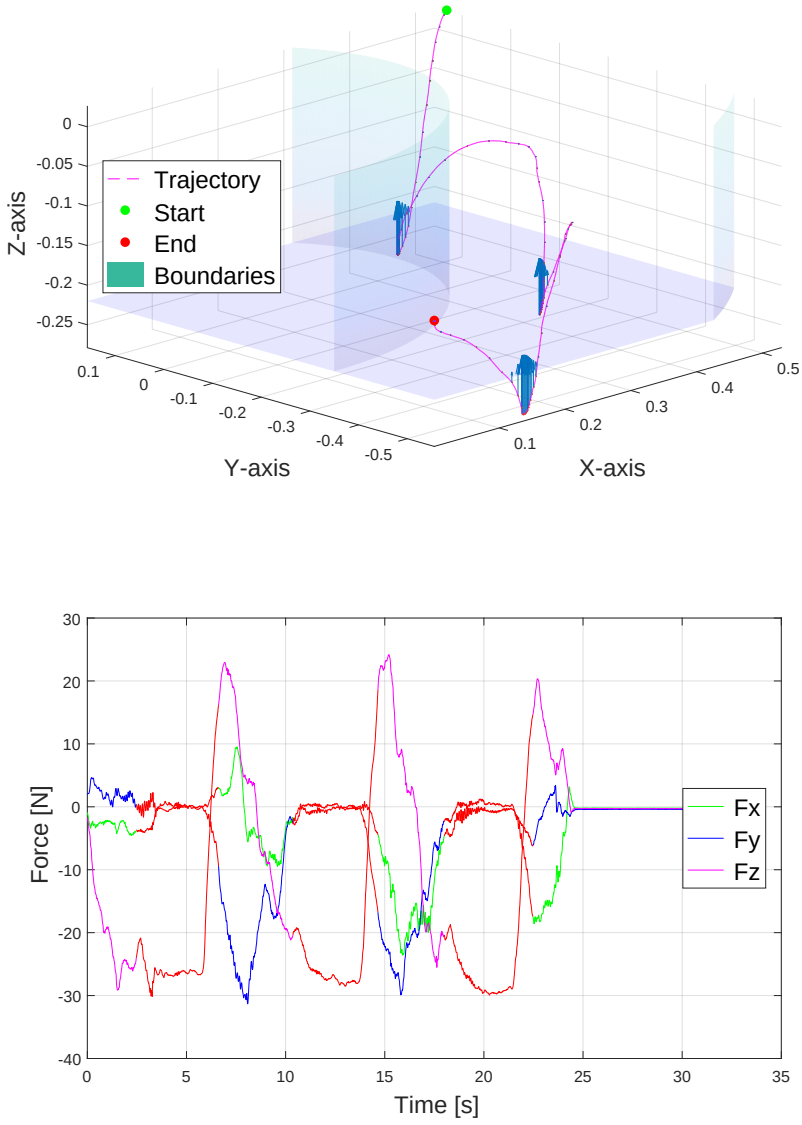


Figure 7.4: Experimental verification test of the lower height soft boundary in the virtual workspace. Force readings turn red when the manipulator is outside the virtual workspace.

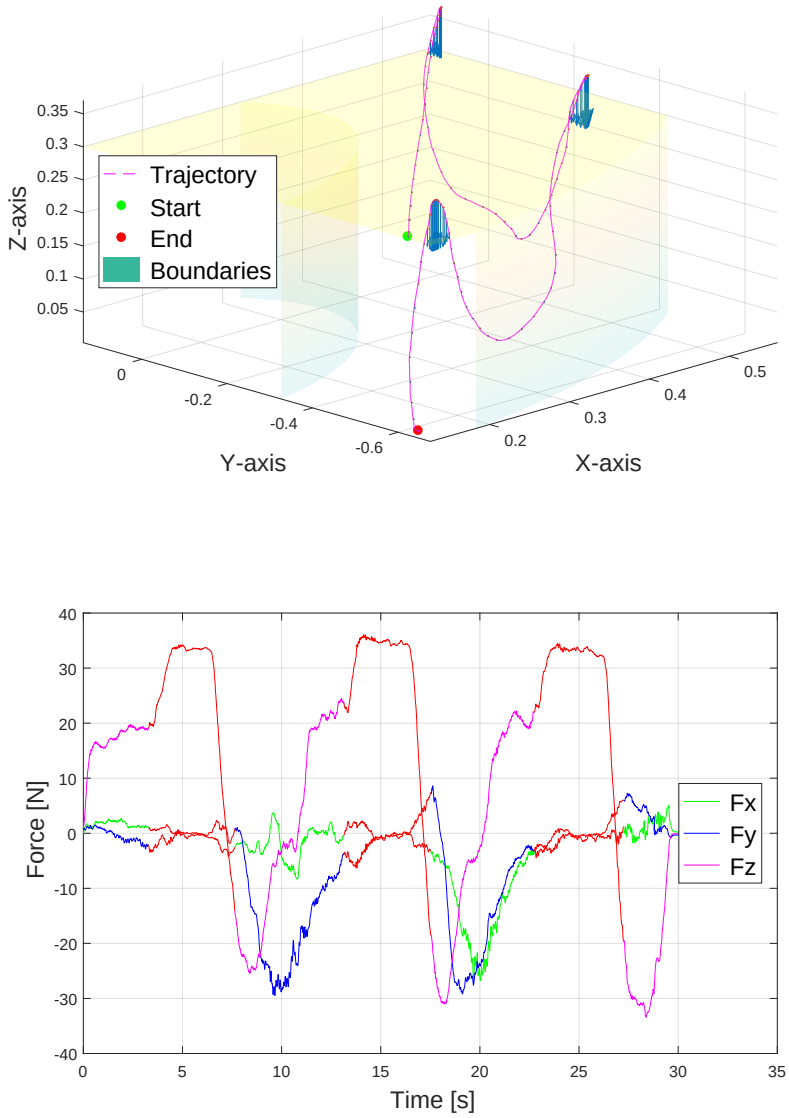


Figure 7.5: Experimental verification test of the upper height soft boundary in the virtual workspace. Force readings turn red when the manipulator is outside the virtual workspace.

Performance analysis

The results illustrated in figure 7.2, 7.3, 7.4 and 7.5 respectively show four experimental verification tests of the outer cylindrical, inner cylindrical, lower height and upper height soft boundaries that constitute the virtual workspace. All four tests penetrate the relevant soft boundary at three different locations. Visual inspection of the TCP trajectories located outside the virtual workspace show radius or height correction vectors that are pointed back towards the virtual workspace with a magnitude that is scaling with the distance from the relevant soft boundary.

The force data results for all four test show complex and varying force readings while moving between the three penetration locations. The three approximate time intervals the TCP spends outside the virtual workspace is $[2.5s - 9.5s]$, $[13s - 18s]$, $[22.5s - 27.5s]$ for the outer boundary test, $[3.5s - 8.5s]$, $[12.5s - 18s]$, $[23s - 27.5s]$ for the inner boundary test, $[2.5s - 6.5s]$, $[10.5s - 14.5s]$, $[18s - 22.5s]$ for the lower boundary test and $[3s - 7.5s]$, $[13s - 17.5s]$, $[22.5s - 27s]$ for the upper boundary with a rounding error accuracy of $\pm 0.5s$. All four test show a stationary force deviation at the peak of the TCP trajectory outside the virtual workspace. The stationary force deviations are consistent with the operator maintaining a position outside the virtual workspace with a constant force for a short time period. The growth rate leading up to the stationary force deviations, particularly visible in figure 7.2, are consistent with the relevant radius- or height correction vectors scaling with the distance from the relevant soft boundary. This effect is easily observable in the force readings in figure 7.2.

7.2 Robotic rehabilitation modes

The four high-level robotic rehabilitation modes presented in chapter 6 needs to be experimentally verified in the laboratory setup. In chapter 2 two ADLs related movements or tasks, functional reach and drinking from a cup, is presented as the suggested experimental verification tests for the robotic rehabilitation system. The simplest robotic rehabilitation mode, 2-plane mode, is not suited for the drinking test as it restricts all vertical movements making the complex drinking test impractical. The drinking test for 2-plane mode is therefore replaced with a circular demonstration test that involves moving the end-effector in a circular path. In chapter 6 it is suggested to restricts movements in random mode that moves close to the face or other fragile body parts as a preventive security measure. The drinking test for random mode is therefore replaced with the same circular demonstration test.

7.2.1 Data and video recording

The results are recorded through numerous data points logged by the external controller at each control iteration for the entire test duration. The TCP trajectory is mapped in Cartesian space with a center-point in the world frame. The force data is recorded in the tool frame and kinematically transformed into the world frame before being presented in the force results. All tests will be performed with the right upper-limb as this is the dominant arm of the operator.

The experimental verification results are complemented with video recordings to better capture the natural interaction between the robot manipulator and a human operator. Unfortunately, the limited access to professional video recording equipment resulted in the videos being recorded after the verification tests. The videos seek to recreate the exact movements presented in the verification results but some deviations must be expected. Links to the relevant videos are presented together with the experimental verification tests, all material is available on GitHub [13] and YouTube [12].

7.2.2 2-plane mode

The robotic rehabilitation 2-plane mode is experimentally verified through a functional reach test[1] and a circular demonstration test[2]. To realistically test the 2-plane mode the operator seeks to constantly apply a heavy force vertically downwards to simulate the added arm weight from a weak stroke patient. The two-dimensional plane of motion is tilted slightly from the horizontal plane to allow for a more natural functional reach interaction with the table. The results are presented in figure 7.6 and 7.7.

[1] https://youtu.be/YtIVcL1GRgw
[2] https://youtu.be/Zctq5M4T-Ag

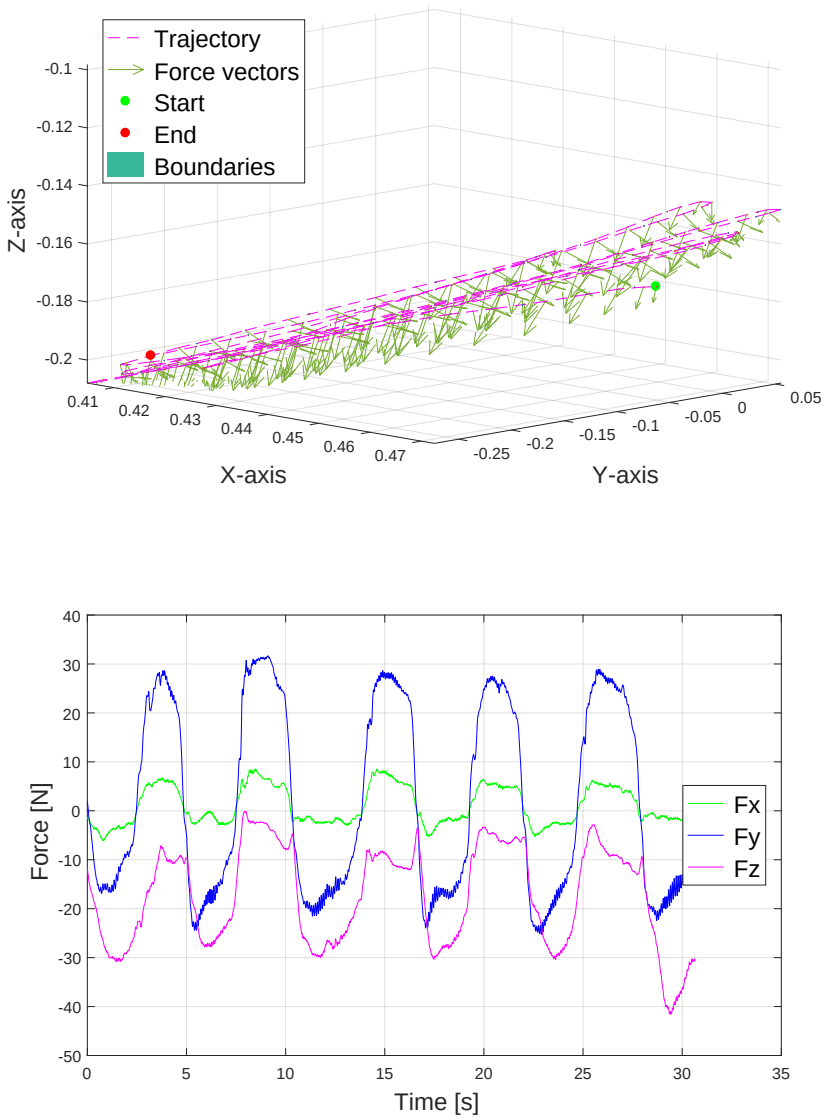


Figure 7.6: Experimental verification test of functional reaches in 2-plane mode. Force readings turn red if the TCP is outside the virtual workspace.

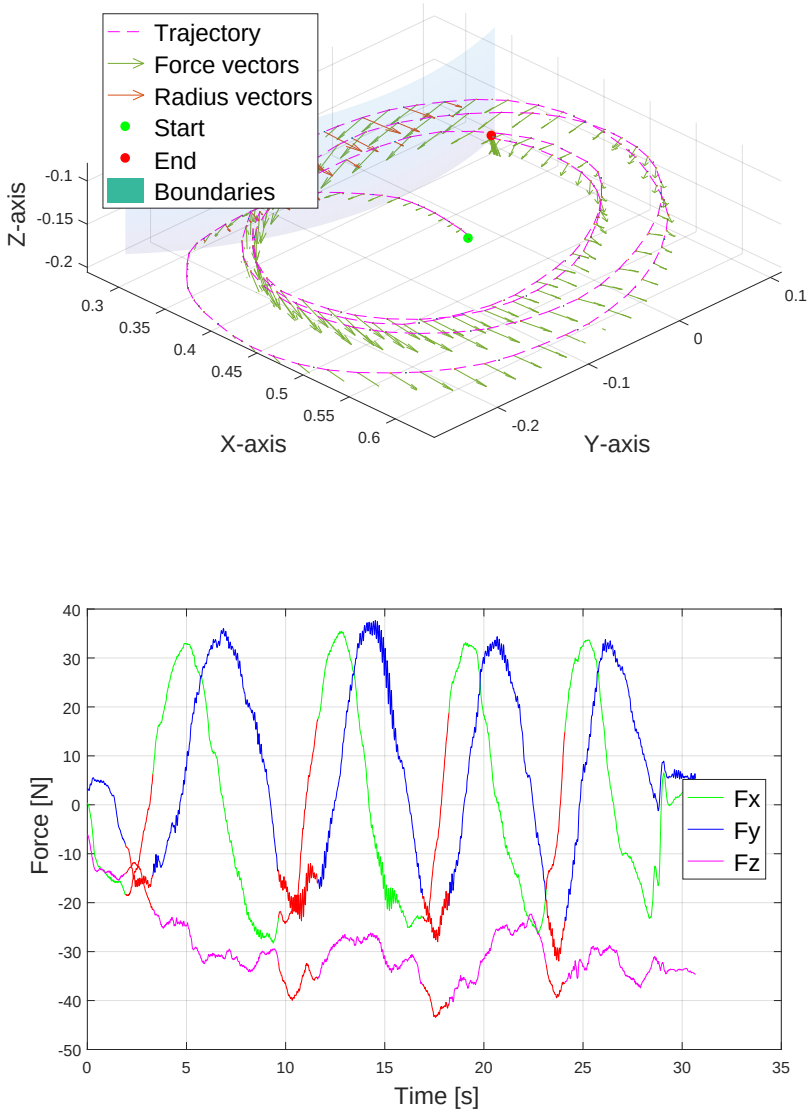


Figure 7.7: Experimental verification test of circular paths in 2-plane mode. Force readings turn red if the TCP is outside the virtual workspace.

Performance analysis

The results illustrated in figure 7.6 show four functional reaches. All four functional reaches are contained within the allowed 2-dimensional plane of motion. The 2-dimensional plane of motion is slightly offset from the horizontal plane. Visual inspection of the trajectory show significant force vectors pointed downwards in a vertical direction. The force readings show a dominating alternating force in F_x and F_y consistent with the upper-limb abduction- and adduction movements that constitute the functional reaches. Observe the consistently negative force in F_z for the entire functional reach test duration.

The results illustrated in figure 7.7 show four circular paths. All four circular paths are contained within the allowed 2-dimensional plane of motion. The 2-dimensional plane of motion is slightly offset from the horizontal plane. Visual inspection of the trajectory show significant force vectors pointed downwards in a vertical direction and that all four circular paths cross the inner cylindrical soft boundary in the virtual workspace. The force readings show a dominating alternating force in F_x and F_y in an approximately sinusoidal behaviour. Both sinusoidal curves have a wave period of $\approx 8s$ were F_y is phase shifted $\approx 1.5s$ relative to F_x. Observe the consistently negative force in F_z for the entire circular demonstration test duration.

7.2.3 Buoyancy mode

The robotic rehabilitation buoyancy mode is experimentally verified through a functional reach test[3] and a drinking test[4] with a constant buoyancy vector magnitude of $15N$. To realistically test the buoyancy mode the operator twists the third wrist joint $\Delta q_6 > 90°$ before initiating an omnidirectional functional reach test. The initial twisting is not performed in the video recording. The omnidirectional functional reach test consists of four consecutive functional reaches; horizontally in the transverse plane, diagonally upwards in the sagittal plane, vertically in the frontal plane and horizontally in the transverse plane respectively. The drinking test objective is to reach for a cup placed on the table with an open hand, firmly grasp the handle, move the cup to the mouth, take a small drink, return the cup to the original placement and return the hand to the starting pose. The results are presented in figure 7.8 and 7.9.

[3] `https://youtu.be/1UK-9a2s99k`
[4] `https://youtu.be/j0s3Y26is_s`

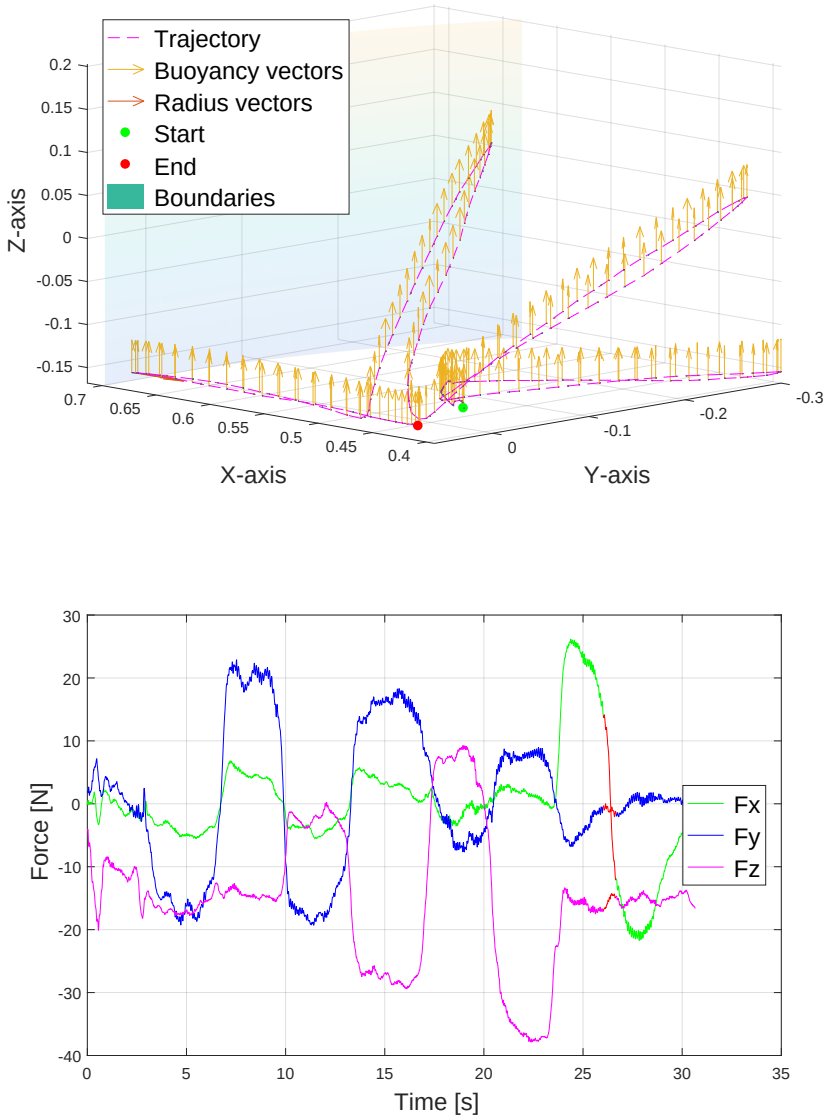


Figure 7.8: Experimental verification test of functional reaches in buoyancy mode. Force readings turn red if the TCP is outside the virtual workspace.

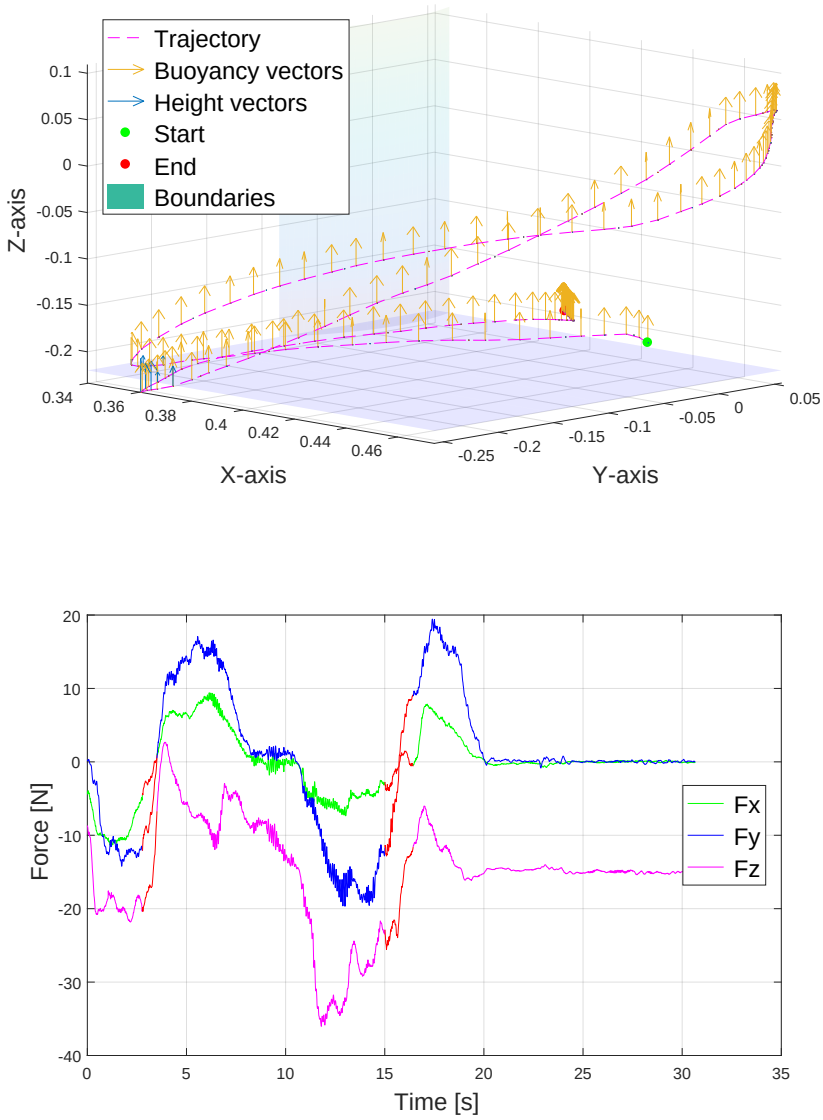


Figure 7.9: Experimental verification test of a drinking task in buoyancy mode. Force readings turn red if the TCP is outside the virtual workspace.

Performance analysis

The results illustrated in figure 7.8 show four omnidirectional functional reaches predominantly located in the XY-plane, XYZ-space, YZ-plane and XY-plane respectively. The first three functional reaches are contained within the virtual workspace, while the fourth functional reach in the XY-plane barely crosses the outer cylindrical soft boundary at the peak of the trajectory. Visual inspection of the trajectory shows yellow buoyancy vectors consistently pointed upwards in a vertical direction and outer radius correction vectors at the peak of the fourth functional reach trajectory. The observation about consistent buoyancy vectors is also valid close to the trajectory starting point associated with the initial rotation of the third wrist joint. The force plot show complex force readings in the approximate time interval $[0s - 3s]$ consistent with the third wrist being twisted. Four dominating force deviations at the approximate time intervals $[3s - 10s]$, $[10s - 17.5s]$, $[17.5s - 23.5s]$ and $[23.5s - 30s]$ with a rounding error accuracy of $\pm 0.5s$ is consistent with the upper-limb abduction- and adduction movements in the XY-plane, XYZ-space, YZ-plane and XY-plane that constitute the four omnidirectional functional reaches. The three first functional reaches are dominated by F_y and F_z components while the fourth functional reach is dominated by F_x and F_z components. Observe that the difference in the F_z magnitude during the third functional reach in the YZ-plane from $[17.5s - 23.5s]$ is consistent with the upper-limb abduction- and adduction movements begin aided and resisted by the buoyancy vectors.

The results illustrated in figure 7.9 show several complex movements connected with the drinking task. The initial functional reach towards the cup moves slightly outside the virtual workspace at the lowest part of the trajectory. The functional reach towards the mouth is followed by a stagnated trajectory while the operator is taking a drink. The cup is placed back on the table with a functional reach that moves slightly outside the virtual workspace. The hand is retracted towards the initial pose by another functional reach that is followed by another stagnated trajectory for the remaining drinking test duration. Visual inspection of the trajectory show buoyancy vectors consistently pointed upwards in a vertical direction, with two small exceptions at the lowest parts of the trajectory where an additional height correction vector is present. The force plot show four dominating force deviations at approximate time intervals $[0s - 3.5s]$, $[3.5s - 8s]$, $[10s - 16s]$ and $[16s - 20s]$ with a rounding error accuracy of $\pm 0.5s$ that is consistent with the four omnidirectional functional reaches. The rotational drinking movement during the approximate time interval $[8s - 10s]$ and the approximate waiting period from $[20s - 30s]$ is consistent with the two stagnated trajectories. Observe the small force vibrations during the rotational drinking movement and the stationary force deviation in F_z in the waiting period.

7.2.4 Resistance mode

The robotic rehabilitation resistance mode is experimentally verified through an omnidirectional functional reach test[5] and a drinking test[6] with a 50% resistance to movement in all six degrees of freedom. The implementation of the functional reach test and the drinking test is similar to the method established in the buoyancy verification test. The difference is that the initial twisting of the third wrist used in the buoyancy verification test is not included in the omnidirectional functional reach test. This initial twisting movement is not relevant for the other robotic rehabilitation modes presented in this master thesis and is therefore not performed. The results are presented in figure 7.10 and 7.11.

[5] https://youtu.be/jLCxZLc3OgU
[6] https://youtu.be/J4RoY8SzUP8

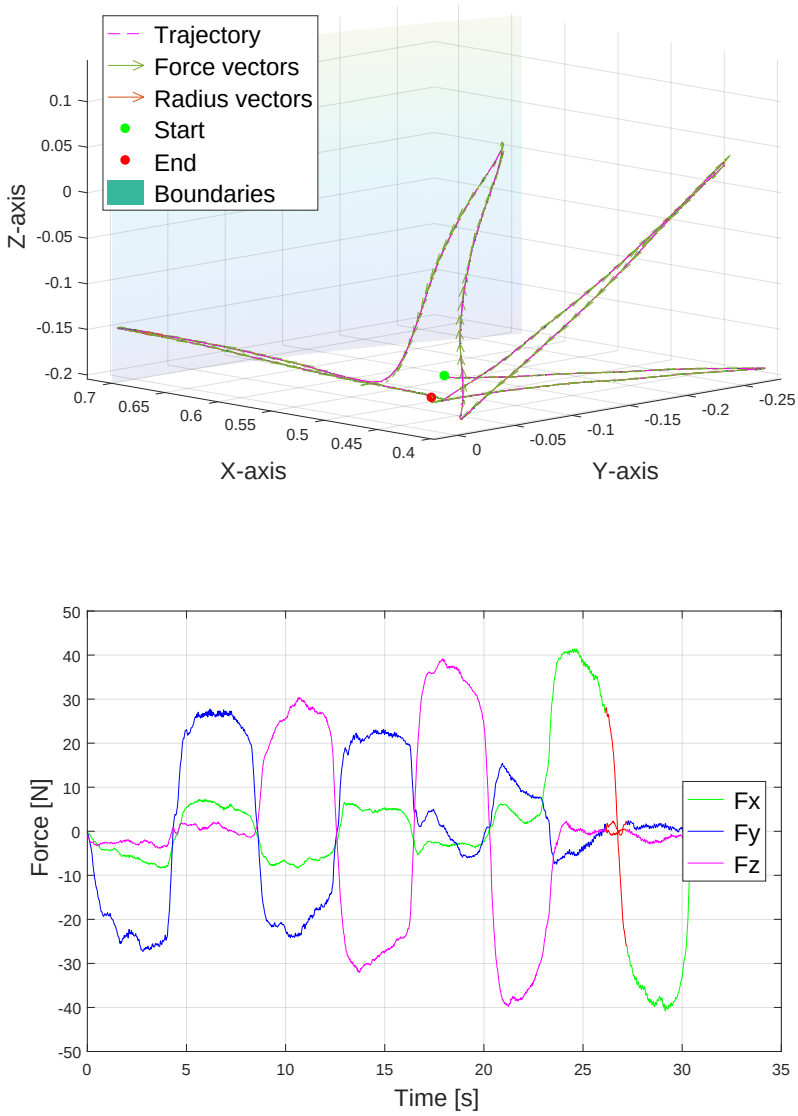


Figure 7.10: Experimental verification test of functional reaches in resistance mode. Force readings turn red if the TCP is outside the virtual workspace.

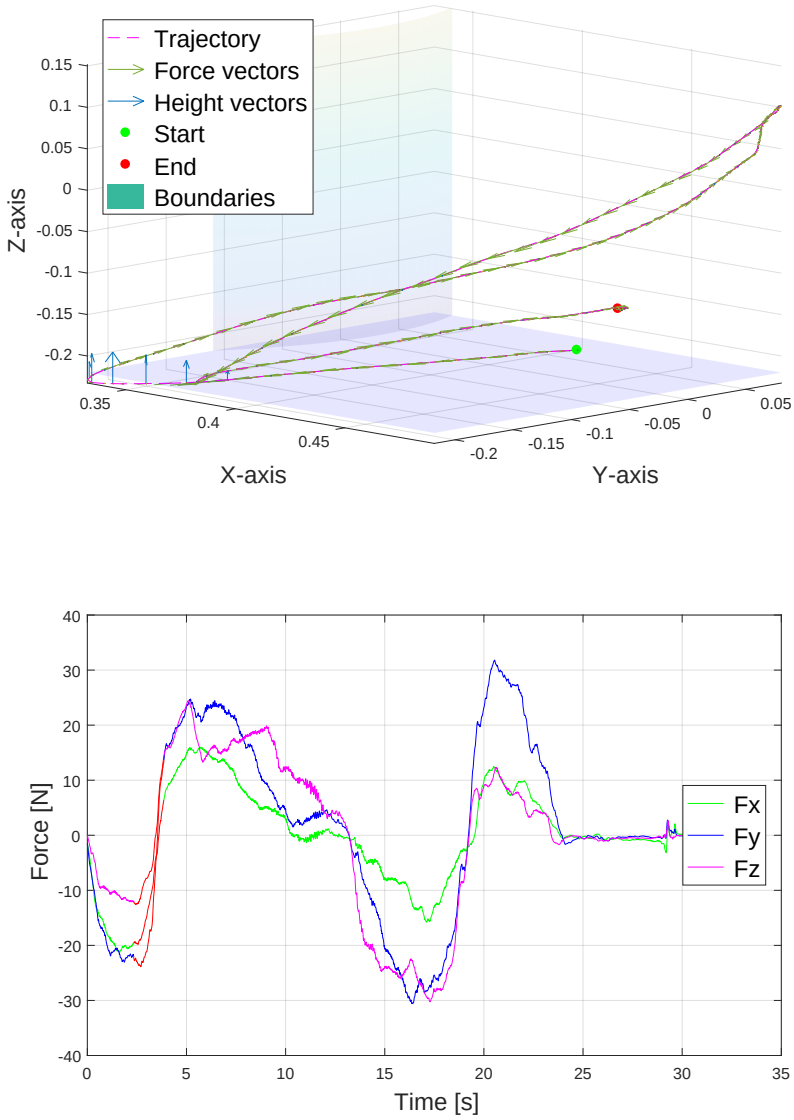


Figure 7.11: Experimental verification test of a drinking task in resistance mode. Force readings turn red if the TCP is outside the virtual workspace.

Performance analysis

The results illustrated in figure 7.10 show four omnidirectional functional reaches predominantly located in the XY-plane, XYZ-space, YZ-plane and XY-plane respectively. The first three functional reaches are contained within the virtual workspace, while the fourth functional reach in the XY-plane barely crosses the outer cylindrical soft boundary at the peak of the trajectory. Visual inspection of the trajectory shows green force vectors consistently pointed along the TCP trajectory and outer radius correction vectors at the peak of the fourth functional reach trajectory. The force readings show four dominating force deviations at approximate time intervals $[0s - 8.5s]$, $[8.5s - 16.5s]$, $[16.5s - 23s]$ and $[23s - 30s]$ with a rounding error accuracy of $\pm 0.5s$ are consistent with the four omnidirectional functional reaches. The three first force deviations consist primarily of F_y and F_z components while the fourth deviation primarily consists of F_x components. Observe the difference in the average magnitude of the force deviations in resistance mode and buoyancy mode. This effect is particularly visible in the fourth functional reach that in buoyancy mode peaks at approximately $26N$ and $-22N$ while the similar fourth functional reach in resistance mode peaks at approximately $41N$ and $-41N$.

The results illustrated in figure 7.11 show complex movements connected with the drinking task. The initial functional reach towards the cup moves slightly outside the virtual workspace at the lowest part of the trajectory. The functional reach towards the mouth is followed by a stagnated trajectory while the operator is drinking. The cup is placed back on the table with a functional reach that is barely contained inside the virtual workspace. The hand is retracted towards the initial pose by a functional reach that is followed by a stagnated trajectory for the remaining drinking test duration. Visual inspection of the trajectory show force vectors consistently pointed along the TCP trajectory, with a small exception at the lowest parts of the TCP trajectory where an additional height correction vector is present. The force readings show four dominating force deviations at time intervals $[0s - 3.5s]$, $[3.5s - 10.5s]$, $[12s - 17s]$ and $[17s - 24s]$ with a rounding error accuracy of $\pm 0.5s$ that is consistent with the four functional reaches. The drinking movement from the approximate time interval $[10.5s - 12s]$ and the approximate waiting period from $[24s - 30s]$ is consistent with the two stagnated trajectories. Observe the small force vibrations during the drinking movement and the lack of a stationary force deviation in F_z in the waiting period.

7.2.5 Random mode

The robotic rehabilitation random mode is experimentally verified through an omnidirectional functional reach test[7] and a circular demonstration test[8] with a random vector magnitude of 100% in all three transitional degrees of freedom. The implementation of the functional reach test is identical to the method established in the resistance verification tests. The implementation of the circular demonstration test is similar to the method established in the 2-plane verification tests, but now the operator seeks to maintain a constant height and a smooth circular path. The results are presented in figure 7.12 and 7.13.

[7] https://youtu.be/Jk4YHPX8J7A
[8] https://youtu.be/ERqwnv1YH2c

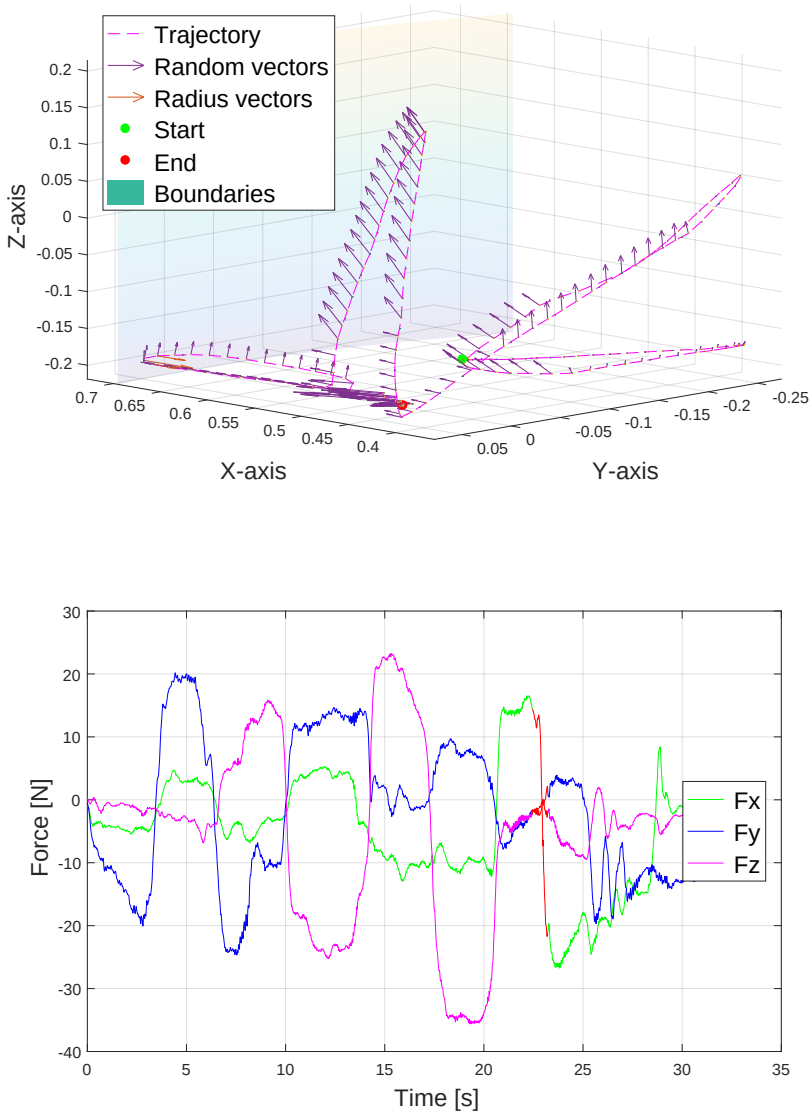


Figure 7.12: Experimental verification test of functional reaches in random mode. Force readings turn red if the TCP is outside the virtual workspace.

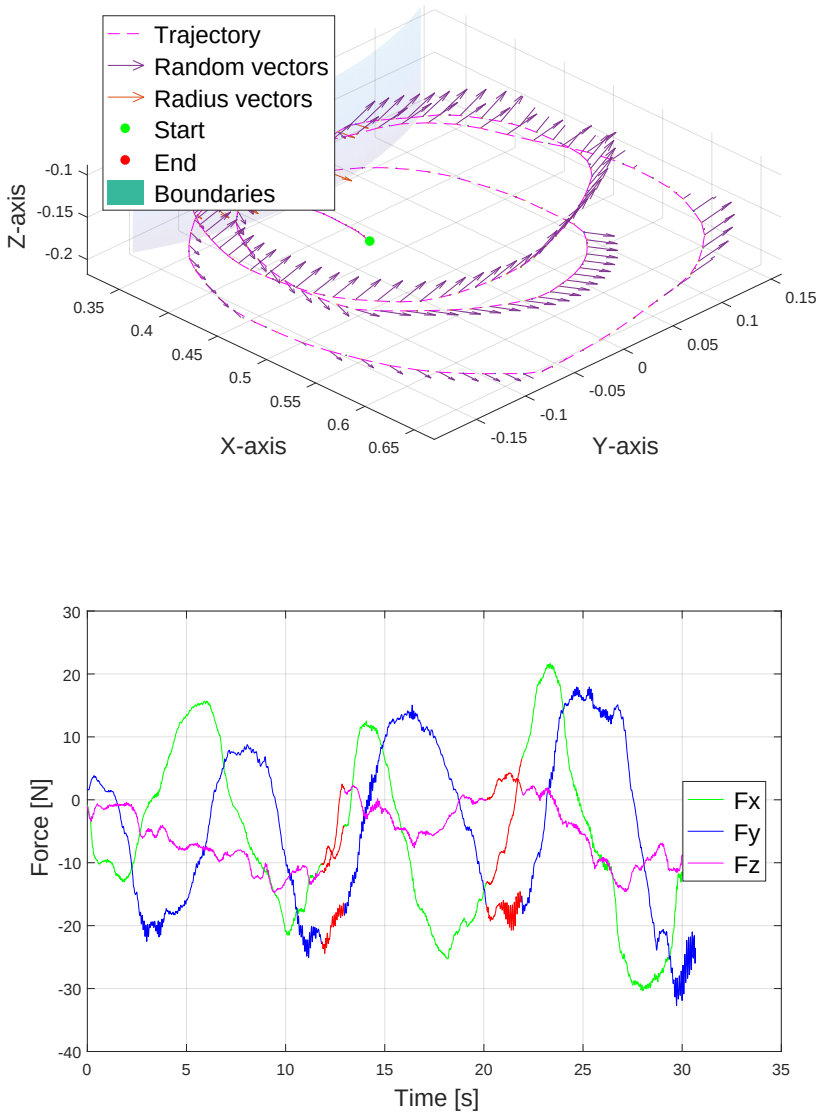


Figure 7.13: Experimental verification test of circular paths in random mode. Force readings turn red if the TCP is outside the virtual workspace.

Performance analysis

The results illustrated in figure 7.12 show four omnidirectional functional reaches predominantly located in the XY-plane, XYZ-space, XYZ-space and XY-plane respectively. Observe the inability to contain the third functional reach to the YZ-plane. The first three functional reaches are contained within the virtual workspace, while the fourth functional reach in the XY-plane barely crosses the outer cylindrical soft boundary at the peak of the trajectory. Visual inspection of the trajectory shows purple random vectors pointed randomly and partially consistently along the TCP trajectory and outer radius correction vectors at the peak of the fourth functional reach trajectory. Parts of the TCP trajectory does not display visible random vectors while the end of the TCP trajectory shows a tight grouping of random vectors with varying magnitude. The force readings show four dominating force deviations at approximate time intervals $[0s - 6.5s]$, $[6.5s - 14.5s]$, $[14.5s - 20.5s]$ and $[20.5s - 25.5s]$ with a rounding error accuracy of $\pm 0.5s$ are consistent with the four omnidirectional functional reaches. The approximate waiting period from $[25.5 - 30s]$ is consistent with the trajectory stagnation close to the end of the TCP trajectory. The three first force deviations consist primarily of F_y and F_z components while the fourth deviation primarily consists of F_x components. Observe the fluctuations in the force readings during the waiting period.

The results illustrated in figure 7.13 show three circular paths. Immediately observe the inability to complete four circular paths within the $30s$ test period. The three circular paths are located in the XYZ-space and are therefore not contained within the intended XY-plane. Two of the circular paths slightly cross the inner cylindrical soft boundary. Visual inspection of the trajectory shows purple random vectors pointed randomly and partially consistently along the TCP trajectory and outer radius correction vectors for the two circular paths that cross the inner cylindrical soft boundary in the virtual workspace. The force readings show a dominating alternating force in F_x and F_y in a behaviour that roughly can be described as sinusoidal. The two sinusoidal curves have an approximate wave period between $9.5 - 10s$ and $8s - 9s$ respectively, were F_y is phase shifted approximately $2.5s - 3s$ relative to F_x with a rounding error accuracy of $\pm 0.5s$. Observe generally high signal noise and the alternating force in F_z for the entire circular test duration. The alternating force in F_z highlights the difficulty in containing the circular movement within the XY-plane.

7.3 Summary

External safety mechanisms

The emergency shutdown procedures are consistently activated when a predefined maximum safety limit is exceeded in all six degrees of freedom. Crossing the soft boundaries in the virtual workspace generates radius- and height correction vectors with a correct direction and a magnitude scaling with the deviation distance. The external security measures and the virtual workspace is correctly implemented and is working as intended.

2-plane mode

The experimental verification tests of 2-plane mode shows that it is possible to choose the desired two-dimensional plane of motion by angling the end-effector into a desired initial pose. It also shows that all further movements are contained within the allowed two-dimensional plane of motion regardless of the amount of force applied to the end-effector. The robotic rehabilitation 2-plane mode is implemented correctly and working as intended.

Buoyancy mode

The experimental verification tests of buoyancy mode shows that it is possible to choose the desired buoyancy force that lifts the stroke patient's arm. The results also show that all further movements are constantly and consistently affected by a correct buoyancy vector regardless of the orientation and position of the end-effector. The behaviour of buoyancy mode is also captured in the tests presented in Appendix B.3. The robotic rehabilitation buoyancy mode is implemented correctly and working as intended.

Resistance mode

The experimental verification tests of resistance mode shows that it is possible to choose the desired resistance to forces and torques in all degrees of freedom. The results also show that all further movements are constantly and consistently affected by a correct resistance to movements regardless of the orientation and position of the end-effector. The behaviour of resistance mode is also captured in the tests presented in Appendix B.3. The robotic rehabilitation resistance mode is implemented correctly and working as intended.

Random mode

The experimental verification tests of random mode shows that it is possible to choose the desired magnitude of the random force vectors generated in the three transitional degrees of freedom. The results also show that all further movements are exposed to random vectors with an erratic behaviour and varying magnitude regardless of the orientation and position of the end-effector. The random movements in the manipulator make following a simple circular path challenging for the operator. The soft step-up implementation of newly generated random vectors prevents large and sudden changes in the manipulator velocity that would be unmanageable and possibly dangerous for stroke patients. The behaviour of random mode is also captured in the tests presented in Appendix B.3. The robotic rehabilitation random mode is implemented correctly and working as intended.

Chapter 8

Discussion

Hundreds of tests of the robotic rehabilitation system are conducted in the laboratory setup. It is not possible within the time-frame of this master thesis to properly document all the tests. Therefore, a subjective assessment of the robotic rehabilitation system together with a discussion of the major implementation challenges faced during the development, will be presented in this chapter. The discussion is centered around the safety of the robotic rehabilitation system and the four high-level control strategies.

8.1 Patient security

Chapter 5 presented three significant implementation challenges affecting the safety of the stroke patients: Unwanted oscillations, mounting bias and workspace limitations.

Unwanted oscillations

The risk associated with unwanted oscillations caused by human arm inertia and spastic symptoms is assessed to be significantly reduced by the control signal filtration, both in terms of probability and consequence. The low-pass filter provided by the F/T sensor introduces a significant increase in system delay. However, it is unknown whether implementation of a low-pass filter from another source can reduce the performance loss while maintaining the required safety to the stroke patient.

Mounting bias

The risk associated with a movement induced by the mounting bias is assessed to be severely reduced by the double software bias, both in terms of probability and consequence. The implementation of the double preventive security measure resulted in reduced performance for the gravity compensation. The reduced gravity compensation performance is solved practically with a dead-band that prevents unwanted drifting in the manipulator. The dead-band solution reduces the overall sensitivity of the system. This

practical dead-band solution does not scale satisfactorily with an increase in equipment weight. The robotic rehabilitation system should be able to handle all use after an initial setup by a technician, as it is not recommended to rely on the users to calibrate the F/T sensor after each reboot or power loss. The implementation challenge with the mounting bias is resolved for practical applications with the current hardware, but future iterations are vulnerable.

Workspace limitations

The large velocity changes in the large base, shoulder and elbow joints induced by moving close to a singularity or the outer spherical boundary of the UR5 workspace is the biggest safety-related implementation challenge. The risk associated with the movements is assessed to be reduced by the virtual workspace. The possibility of crushing damage is still present in the smaller joints, but this requires an intentionally bad and artificial trajectory in the end-effector. The probability for this to occur during normal operation is assessed to be very low. To further discourage the unwanted behaviour, an initial end-effector pose with a good range of motion in all joints and natural interaction with the table is chosen for all robotic rehabilitation modes. The potential consequence of a self-collision with the smaller wrist joints is reduced due to the lower momentum involved. This makes it is easier for the internal- or external controller to detect and stop the movement before any significant damage occurs. Future iterations of the robotic rehabilitation system implemented on a UR5 robot that is running the latest 3^{rd} generation internal controller can safely rely on the user-generated safety parameters inside the Polyscope GUI to limit the UR5 workspace in a similar manner.

8.2 Robotic rehabilitation modes

8.2.1 Compliance mode

The compliance mode is perceived as a low-level control strategy that is accurate and stable. The mode is designed to be a foundation for the other robotic rehabilitation modes and as a general development tool. The mode is not applicable in traditional stroke rehabilitation exercises but is could function as an introduction and demonstration of the robotic rehabilitation system for new stroke patients unfamiliar with the system.

8.2.2 2-plane mode

The recommendation in section 2.1.2 cautioned about implementing constraint-induced movement therapy, mainly based on the lack of scientific documentation on the positive effect of constraint-induced movement therapy on stroke patients. The design restrictions imposed on prototype 1.1 is a direct consequence of the recommendation, as accidentally implementing constraint-induced movement therapy on the entire robotic rehabilitation system is undesired. The robotic rehabilitation 2-plane mode is designed to challenge the established view and provide a basis for further scientific documentation on constraint-induced assistive movement therapy on stroke patients.

The 2-plane mode is designed to be generally limiting and is applicable in specialized exercises in traditional stroke rehabilitation exercises. The mode is perceived as a high-level control strategy that is not engaging with an interaction that stagnates quickly. This impression is probably amplified in a healthy individual, but would still apply to stroke patients. The behaviour is consistent with the intended design, but for a healthy individual the interaction with the manipulator quickly become uninteresting.

8.2.3 Buoyancy mode

The buoyancy mode implementation is vulnerable to a twitching behaviour caused by the rapid updates in the direction of the buoyancy vector. This is particularly prevalent during rapid rotational changes in the tool frame. The twitching is not directly dangerous to the operator or patients, but it is unwanted as it can easily be perceived as dangerous by the patients. Increasing the buoyancy vector update rate is not possible since the UR5 robot is running with a maximum updating rate, but decreasing the total system delay could allow for a more accurate directional update. It is possible that this solution could eliminate the unwanted twitching. Increasing the cut-off frequency in the low-pass filter is the only available variable that could achieve the desired decrease in system delay. However, these changes to the low-pass filtration introduce control signal noise that induces unwanted oscillations. Compromising between the two factors is possible, but changes in the low-pass filtration effect the entire system while the buoyancy vector update rate is isolated to a single robotic rehabilitation mode. Another solution is implementing control signal filtration with the ability to predict the next update to the buoyancy vector. The prediction allows for a preemptive update to the buoyancy vector direction that could eliminate the challenge with oscillations. Predicting human movements is however defined as being outside the scope of this master thesis. The inability to spend more time on control filtration leads to the practical and simple solution to maintain the original low-pass filtration and be mentally aware to avoid rapid rotations in the end-effector during while utilizing buoyancy mode.

Buoyancy mode is designed to be highly dynamic and applicable in all conceivable traditional stroke rehabilitation exercises. The mode is perceived as a high-level control strategy with high potential, but it is susceptible to unwanted twitching in the manipulator. The simple functionality is a highly useful rehabilitation tool for a physical therapist.

8.2.4 Resistance mode

Early testing and development in resistance mode revealed a weakness in the drinking test. An important motivation behind the design of prototype 1.1 is maintaining the patients grasping functionality. Unfortunately, the physical interactions between the aluminum ring and the cup handle generated small unwanted twitching in the manipulator. The twitching is not directly dangerous to the operator or patients, but it is unwanted as it can easily be perceived as dangerous by the patients. The underlying challenge with unwanted twitching caused by unintended control inputs generated from vibrations between the two stiff

objects needs so be properly addressed by a re-design of the end-effector or through additional control signal filtration. Being unable to spend more time on end-effector design or control signal filtration the simple solution is to avoid direct contact by placing one or two fingers between the aluminum ring and the cup handle during the drinking tests for the two relevant modes.

Resistance mode is perceived as the most stable high-level control strategy as it inadvertently resists any unwanted oscillations and twitching in a similar manner as the intended resistance to patient-generated movements. The mode is designed to be highly dynamic and applicable in all conceivable traditional stroke rehabilitation exercises.

8.2.5 Random mode

The preventive security measure presented in section 6.4 restrict the application of random mode to tasks or movements that avoid moving close to the face or other delicate body parts. This safety-oriented decision should be maintained for future iterations of the robotic rehabilitation system. The soft step-up implementation to the magnitude in newly generated random vectors is designed to prevent rapid velocity changes in the end-effector. The experimental and subjective testing show that the soft step-up is working as intended, it is best described as a gentle but firm push. Despite the security measures working as intended the mode is still perceived as prone to trigger spastic muscle contractions and stroke patients with spastic symptoms should be cautious with utilizing random mode. The final assessment on whether stroke patients with spastic symptoms should utilize random mode is viewed as a discretionary assessment that should be performed by professional physical therapists working together with the individual patient.

Random mode is perceived as the most engaging and challenging high-level control strategy. The mode is designed to be highly dynamic and applicable in a security oriented selection of traditional stroke rehabilitation exercises. The interaction with the manipulator is naturally highly unpredictable and the added challenge to perform any task or movement is perceived as consistently interesting and motivating.

Chapter 9

Conclusion and Future work

9.1 Conclusion

The initial assignment analysis explains the implementation approach towards solving the thesis goals and objectives. Different rehabilitation strategies are presented and discussed together with two suggested experimental verification tests based on ADL, a drinking test and a functional reach test. The intended implementation of a control feedback loop in tool space to encourage the natural human-machine interaction is presented. The need for an external force/torque sensor is discussed. The precondition of utilizing the UR5 manipulator from Universal Robot is discussed, the Mini45 force/torque sensor from ATI is identified as suitable external sensor and discussed.

The chosen hardware is then integrated into a laboratory setup. The F/T sensor is mounted on the end-effector and three prototypes for the physical human-machine interface is developed, analyzed and tested. Investigations into the available software reveal that the URScript based ur_modern_driver provides the best basis for a low-level control strategy by streaming speedj commands to the internal controller. The entire hardware and software architecture for the robotic rehabilitation system is presented. The system delay is estimated through documentation and experimental testing.

The desired control loop implemented directly in tool space require a mathematical model of the UR5. The forward- and inverse kinematic is presented together with the DH parameters for the UR5. The inverse kinematics is briefly presented. The inverse velocity kinematics is presented as a method to realize the desired joint velocity based control interface.

The best-suited F/T sensor sampling frequency of $250Hz$ is presented and discussed relative to the stroke patient consequences, human reaction time and hardware limitations. Three implementation challenges directly relevant to the safety of the stroke patients is presented, analyzed and discussed. The implemented solutions require low-pass filtration with a $5Hz$ cut-off frequency, double software bias to the F/T mounting bias and a limiting

virtual workspace. The solutions are presented together with a brief analysis of the consequences to system performance. An overview of the system security achieved through the internal- and external safety mechanisms are presented.

The gravity compensation is negatively affected by one of the implemented solutions to increase the stroke patients security. The gravity compensation implemented with an artificial gravity vector and a dead-band filtration with intervals $\forall F = \pm 1N, \forall \tau = \pm 0.5Nm$ is compromised, but functional for practical applications. The systems executive controller design is presented with an unknown internal controller. The design of the external controller is therefore thoroughly and incrementally developed and tested through experimental tuning and subjective testing. The best possible performance is achieved through splitting the force and torque control inputs into two parallel P-controllers with proportional gain parameters $K_p = 0.005$ and $K_{p,T} = 0.4$ respectively. Four high-level control strategies for robotic rehabilitation of upper-limbs in stroke patients is presented based on the low-level control strategy.

The safety of the robotic rehabilitation system is experimentally verified through emergency shutdown tests and soft boundary tests. The behaviour of the four high-level control strategies named 2-plane mode, buoyancy mode, resistance mode and random mode is experimentally verified by the two suggested verification tests. The analysis or raw data is complemented by video recordings of the human-machine interaction.

The accumulated experience from hundreds of documented and undocumented tests is presented and discussed. The discussion naturally centers around the system security and the high-level control strategies.

The overall safety provided by the robotic rehabilitation system with the current hardware through a combination of external safety mechanisms and internal safety mechanisms is perceived as satisfying, consistent and suitable for further development.

The work presented in this master thesis show that it is feasible to provide robotic rehabilitation of upper-limbs in stroke patients by utilizing a standard 6-DOF industrial robotic manipulator combined with an external force/torque sensor.

9.2 Future work

The results presented by this master thesis is promising and hopefully a foundation for future research and development into robotic rehabilitation of upper-limbs in stroke patients. This section seeks to describe suggested focus areas for future development of the robotic rehabilitation system.

9.2.1 End-effector design

The end-effector prototype 1.1 presented in figure 3.7 needs further development. The rudimentary physical interface is mostly utilized with a glove to ensure a tight fit inside the aluminum ring. The wristband introduces slack to the patient-generated control signal. The twitching caused by interaction with the aluminum ring and grasped objects is unwanted. The next prototype iteration should seek to provide a custom-made physical interface to the stroke patients. The prototype should be easily interchangeable with a quick release interface to the end-effector. This can be achieved through molding a prosthesis and co-operating with physical therapists and orthopedic specialists.

Another area of improvement is incorporating a hardware-based security measure into the end-effector design. The main idea is to have a physical connection that breaks of or gives way when forces and torques exceed safe levels. Work towards this security redesign was initiated but time did not allow for the production of a finished prototype design. Please refer to the main supervisor Øyvind Stavdahl, MSc, Ph.D. for the original idea and further details.

9.2.2 Control signal filtration

The low-pass filtration and dead-band filtration presented in section 5 negatively affect system performance in different ways in order to increase the security of the system. The low-pass filtration increases the system delay and the dead-band filtration lowers the sensitivity. Investigating whether other implementations of the control signal filtration can reduce the negative effects while retaining the security level is a possible future research area.

Additionally, implementation challenges with twitching due to rapid updates in reference vectors could be solved with additional control signal filtration able to predict future vector updates. Possible methods for achieving this is Kalman filtration or similar filtration methods that require either a mathematical model of a human arm, human movement or filtering based on a predetermined path.

9.2.3 GUI

The robotic rehabilitation system currently interacts with the users in a Linux environment through a terminal interface. This GUI is rudimentary and is not suited for the intended users. It should be replaced with a more professional and user-friendly GUI.

Reworking the GUI also allow for an interactive interface with the patients during rehabilitation exercises. Improving the interaction with the users and stroke patients can improve the overall experience and patient motivation towards rehabilitation.

9.2.4 Rehabilitation exercises

The robotic rehabilitation system seeks to be a useful rehabilitation tool for physical therapists. In order to be useful, it is important that the user understands the potential of the tool. Additional rehabilitation exercises suitable for the different modes in the robotic rehabilitation system should be identified. The additional exercises should be experimentally verified in a laboratory setup.

9.2.5 Expanding the system

The external controller can easily be expanded to implement with other robotic rehabilitation modes then the four modes designed and implemented in this master thesis. The results from the low-level control strategy experimental verification can be used as a basis for future iterations of the robotic rehabilitation system based on the UR5 from Universal Robots.

Smaller changes in the code allow for the implementation of the robotic rehabilitation system on a UR3 or UR10. The low-level controller probably requires an experimental re-tuning, the safety mechanisms and kinematic parameters needs to be changed to fit the new environment. The external force/torque sensor can also be replaced with minimal changes in the code as long as similar control filtration techniques are implemented and experimentally verified.

Bibliography

[1] Maria Fernanda Soares de Almeida Guilherme Lopes Cavalheiro Ana Paula Souza Paixão Sheila Bernardino Fenelon Adriano O. Andrade, Adriano Alves Pereira and Valdeci Carlos Dionisio. Human tremor: Origins, detection and quantification. http://cdn.intechopen.com/pdfs/41790/InTech-Human_tremor_origins_detection_and_quantification.pdf. (Accessed on 02/27/2017).

[2] Thomas Timm Andersen. *Optimizing the Universal Robots ROS driver.* Technical University of Denmark, Department of Electrical Engineering, 2015.

[3] Thomas Timm Anderson. Sorce code for ur_modern_driver. https://github.com/ThomasTimm/ur_modern_driver, Desember 2016.

[4] ATI Industrial Automation. F/t sensor mini45. http://www.ati-ia.com/products/ft/ft_models.aspx?id=Mini45. (Accessed on 04/11/2017).

[5] ATI Industrial Automation. Mini45-e image (2140×1608). http://www.ati-ia.com/app_content/product_images/Mini45-E%20low_res.jpg. (Accessed on 04/11/2017).

[6] Kristine Blokkum. Design and analysis of a hybrid position/force controller for robotic rehabilitation of upper limb after stroke. Master's thesis, Norwegian University of Science and Technology, Department of Engineering Cybernetics, 2016.

[7] Kristine Blokkum. Functional review on robotic rehabilitation of upper limb after stroke, 2016. Project thesis, Norwegian University of Science and Technology, Department of Engineering Cybernetics, 2016.

[8] Faculty of Medicine NTNU Department of Neuroscience. Bev3103 - movement control and motion problems. http://www.ntnu.edu/studies/courses/BEV3103, October 2016.

[9] GNU Project-Free Software Foundation. Gsl - gnu scientific library. https://www.gnu.org/software/gsl/. (Accessed on 06/06/2017).

[10] Kelsey P. Hawkins. Analytic inverse kinematics for the universal robots ur-5/ur-10 arms. https://smartech.gatech.edu/bitstream/handle/1853/50782/ur_kin_tech_report_1.pdf, December 7 2013. (Accessed on 06/06/2017).

[11] Helsedirektoratet. Nasjonal retningslinje for behandling og rehabilitering ved hjerneslag. *ISBN-978-82-8081-153-0*, 2010.

[12] Mads Johan Laastad. Youtube channel. https://www.youtube.com/channel/UCe7ghbSmB6slRAFVm__W9_w. (Accessed on 06/06/2017).

[13] Mads Johan Laastad. Github account. https://github.com/madlaa/NTNU-master-thesis, 2017. (Accessed on 06/05/2017).

[14] Mads Johan Laastad. Robotic rehabilitation of upper limb after stroke, 2017. Project thesis, Norwegian University of Science and Technology, Department of Engineering Cybernetics, 2017.

[15] P. Maciejasz. A survey on robotic devices for upper limb rehabilitation. *J Neuroeng Rehab. R&D*, 2014.

[16] Hocoma media center. Example image of exoskeleton utilized in robotic stroke rehabilitation. https://www.hocoma.com/uploads/pics/1508_armeopower.jpg, December 2016.

[17] Lofthus & Meyer. Consequences of hip fracture on activities of daily life and residential needs., 2004.

[18] Christopher Murray and Alan Lopez. The world health report 2002 - reducing risks, promoting healthy life, 2002.

[19] N. Norouzi-Gheidari, P.S. Archambault, and J. Fung. Effects of robot-assisted therapy on stroke rehabilitation in upper limbs: Systematic review and meta-analysis of the literature. *J. of Rehab. R&D*, pages p. 479–495, 2012.

[20] Prestmo. The trondheim hip fracture trial., 2015.

[21] Industrial robot. Teach pendant image. http://industrialrobotinfo.apps-1and1.net/wp-content/uploads/2015/04/ur_teach_pendant_moving-co.jpg, December 2016.

[22] Universal Robot. Manual for ur5 cb2 v.1.6. https://s3-eu-west-1.amazonaws.com/ur-support-site/17955/manual_en_UR5_CB2_1.6.pdf. (Accessed on 04/11/2017).

[23] Universal Robot. Ur5 user manual global. https://s3-eu-west-1.amazonaws.com/ur-support-site/27421/UR5_User_Manual_en_Global.pdf. (Accessed on 04/04/2017).

[24] Universal Robots. Service manual for ur5 with cb3.0/cb3.1-controller. https://www.universal-robots.com/download/?option=15833#section15832, December 2016. See figure 3.1.2 on page 11.

[25] Universal Robots. Service manual for ur5 with cb3.0/cb3.1-controller - limiting safety functions. https://www.universal-robots.com/download/?option=15833#section15832, December 2016. See table on page I-14.

[26] Foundation GNU Project Free Software. Gsl - gnu scientific library. https://www.gnu.org/software/gsl/. (Accessed on 05/23/2017).

[27] Luciana B. Sollaci and Mauricio G. Pereira. *The introduction, methods, results, and discussion (IMRAD) structure: a fifty-year survey.* Journal of the Medical Library Association, 2004.

[28] Mark W. Spong & Seth Hutchinson & M. Vidyasagar. *Robot modeling and control*. John Wiley & Sons, Inc., 2006.

[29] Hakan Yildirim. Robot control using vision and force sensors. Master's thesis, Technical University of Denmark, Department of Electrical Engineering, 2016.

[30] Zacobria. Image of complete ur5 system (2790×2212). http://www.zacobria.com/images/robot_complete.jpg. (Accessed on 04/11/2017).

Appendix A

Source code

This Appendix briefly describes the source code and implementation files. The description assumes a fundamental understanding of the Linux environment and code compilations. The entire code repository for the robotic rehabilitation system is available on GitHub[1] [13] or by contacting the Department of Engineering Cybernetics at NTNU.

Repository file structure

```
NTNU-master-thesis/                  -- main repository
    MATLAB-data-analysis/            -- MATLAB code and figures
    data/                            -- raw log files
        log_library/
        logs/
    force/                           -- external controller
        force.cpp
        force.h
        ujac_c.c
    kinematics/                      -- mathematical tools
        ur_kin.cpp
        ur_kin.h
        ikfast.h
    rrulas/                          -- GUI and initial setup
        Makefile
        rrulas.cpp                   -- main file
        rrulas.h
        start.exe
    ur_modern_driver-master/         -- driver for UR robots
    video/                           -- videos from experiments
    README.md                        -- README
```

[1] https://github.com/madlaa/NTNU-master-thesis

File description

- **force.cpp & force.h:** These files contain the external controller and safety mechanisms developed and implemented during this master thesis. Parts of the framework seeks to re-use code developed by [29].
- **ujac_u.c:** This file is the C implementation of the Jacobian for the UR5 developed by [29]. The DH parameters can be changed in order to fit the Jacobian for the UR3 or UR10.
- **ur_kin.cpp & ur_kin.h & ikfast.h:** These files contain the implementation of the forward- and inverse kinematics for the UR5 and UR10 developed by [10].
- **Makefile & start.exe:** These files contain the makefile and executable file for the robotic rehabilitation system developed during this master thesis.
- **rrulas.cpp & rrulas.h:** These files contain the main file, various robot motions and user interface for the robotic rehabilitation system developed during this master thesis. Parts of the framework seeks to re-use code developed by [29].
- **ur_modern_driver-master/:** New drivers for the UR3, UR5 and UR10 from Universal Robot developed by [3]. The driver documentation is developed by the same author [2].

Running the code

The robotic rehabilitation system needs a non-standard Linux library before it can be utilized. The GNU Scientific Library [9] is utilized to calculate the inverse Jacobian. Follow the installation instructions and update the library before proceeding.

Other essential preparations require recreating parts of the laboratory setup utilized in this master thesis.

- Mount the F/T sensor Mini45-E transducer with the sensor frame identical to the tool frame, if not make sure to compensate for the difference in the external controller. Utilize the F/T Net Box web interface to configure the Mini45 F/T sensor with a $250Hz$ sampling frequency, low-pass filtration with a cut-off frequency of $5Hz$ and remove the F/T mounting bias.
- Clone the master thesis repository from GitHub or contact the Department of Engineering Cybernetics at NTNU to obtain a copy of the source code.
- The IP addresses of the UR5 and F/T sensor are defined in rrulas.cpp and force.cpp respectively. Make sure to change them appropriately to fit your laboratory setup.
- The code now needs to be recompiled. Enter the directory NTNU-master-thesis/rrulas and run the terminal command *"make clean && make"*.

username@ubuntu:~/NTNU-master-theis/rrulas$ make clean && make

- Be aware and make the workspace available for the predetermined initial pose the manipulator moves to when the system is executed. The initial TCP position will be approximately $[X, Y, Z] = [0.4717m, -0.0589m, -0.1677m]$ in Cartesian coordinates relative to the base frame. The external controller is not active during this initial movement.

- The robotic rehabilitation system can now be executed with the newly compiled executable start.exe file with the terminal command *"./start"*.

  username@ubuntu:~/NTNU-master-theis/rrulas$ ./start

- Always keep the teach pendent with the emergency button available while running the robotic rehabilitation system. Respect the power of the UR5 manipulator and do not misuse the robotic rehabilitation system.

Appendix B

Miscellaneous

B.1 Universal Robot UR5

B.1.1 Denavit-Hartenberg convention

The Denavit-Hartenberg (DH) convention can simplify calculations by reducing the number of variables used to describe an arbitrary homogeneous transformation matrix from six to four parameters. This is achieved by assigning reference frames to each joint based on a set of predefined criteria;

DH1 The axis x_i is perpendicular to the axis z_{i-1}.

DH2 The axis x_i intersects the axis z_{i-1}.

The four resulting DH parameters are defined as,

θ_i: angle with positive sense from x_{i-1} to x_i measured in a plane normal to z_{i-1}.

d_i: distance from the origin o_{i-1} to the intersection of the x_i axis with z_{i-1} measured along the z_i axis.

a_i: distance between the axes z_{i-1} and z_i measured along the x_i axis.

α_i: angle with positive sense from z_{i-1} to z_i measured in a plane normal to x_i.

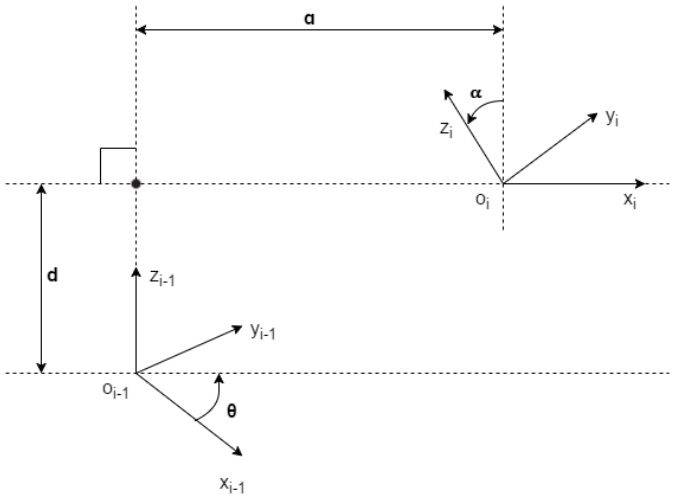


Figure B.1: DH convention for positive sense of angles θ and α.

B.1.2 Reference frames for UR5 robot manipulator

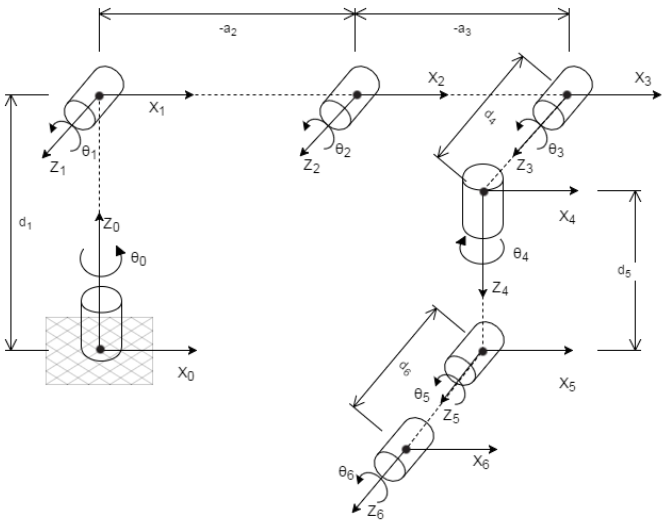


Figure B.2: Reference frames for UR5 based on Denavit-Hartenberg convention.

B.1.3 URScript motion commands

Command	Description
movec	Move circular to a position in Cartesian space
movej	Move linear to a position in joint space
movel	Move linear to a position in Cartesian space
movep	Blend circular and move linear to a position in Cartesian space
servoc	Servo circular to a position in Cartesian space
servoj	Servo linear to a position in joint space
speedj	Set joint speeds
speedl	Set tool speed

Table B.1: Motion commands from the URScript programming language.

B.1.4 Configuration of the UR5 robot manipulator

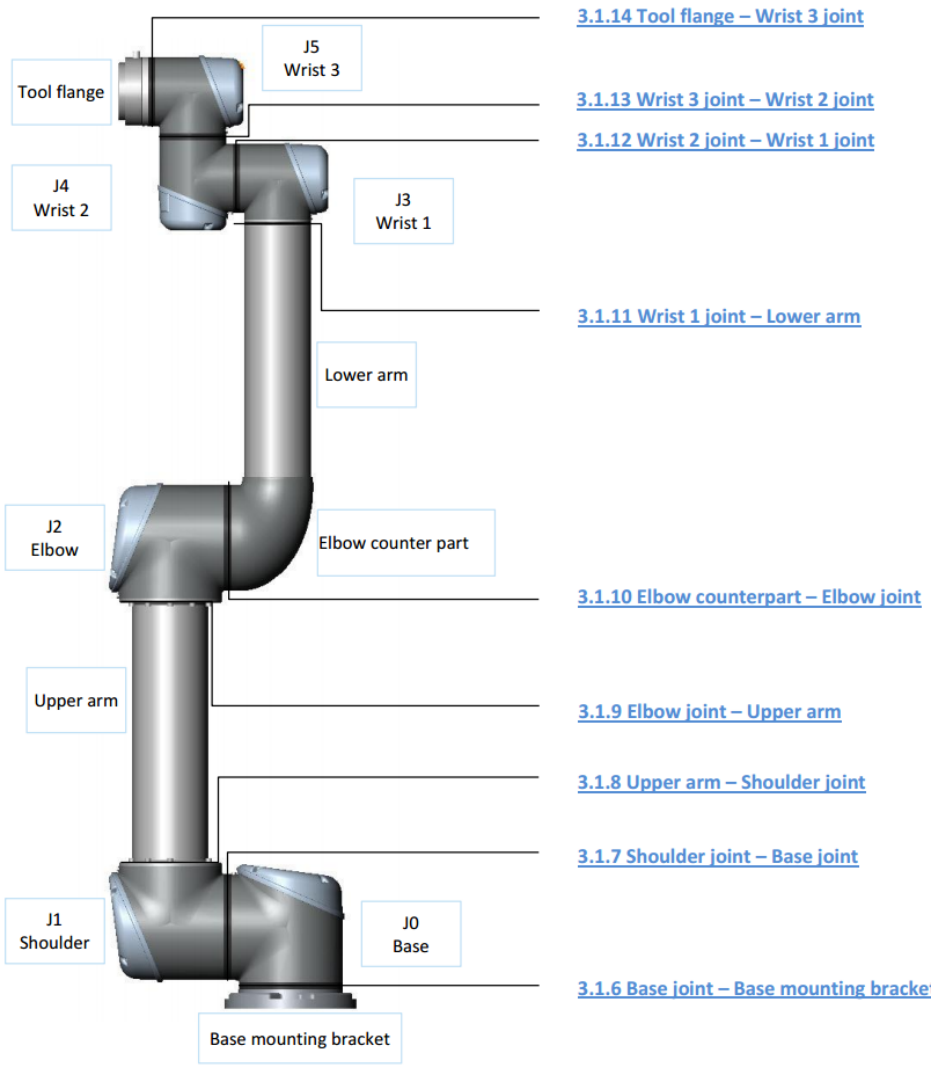
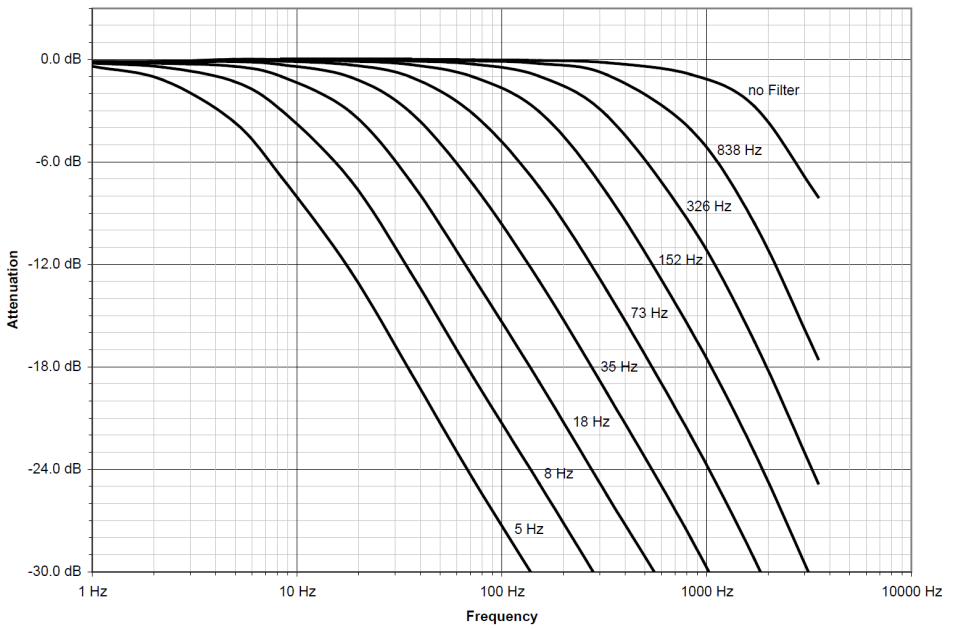
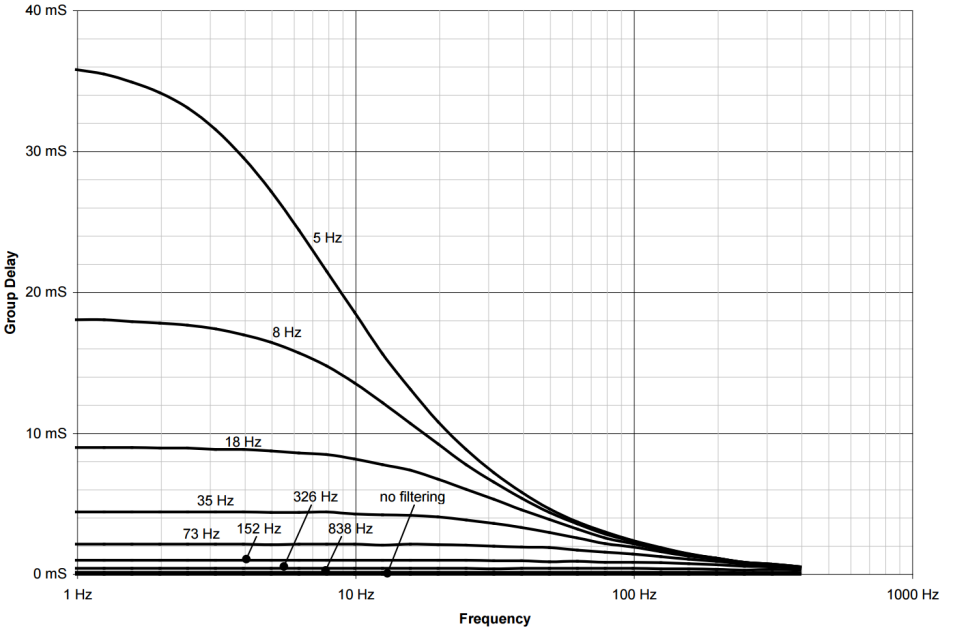


Figure B.3: Configuration details of the UR5 robot. Figure taken from [24]

B.2 Low-pass filtration in the Mini45 F/T sensor



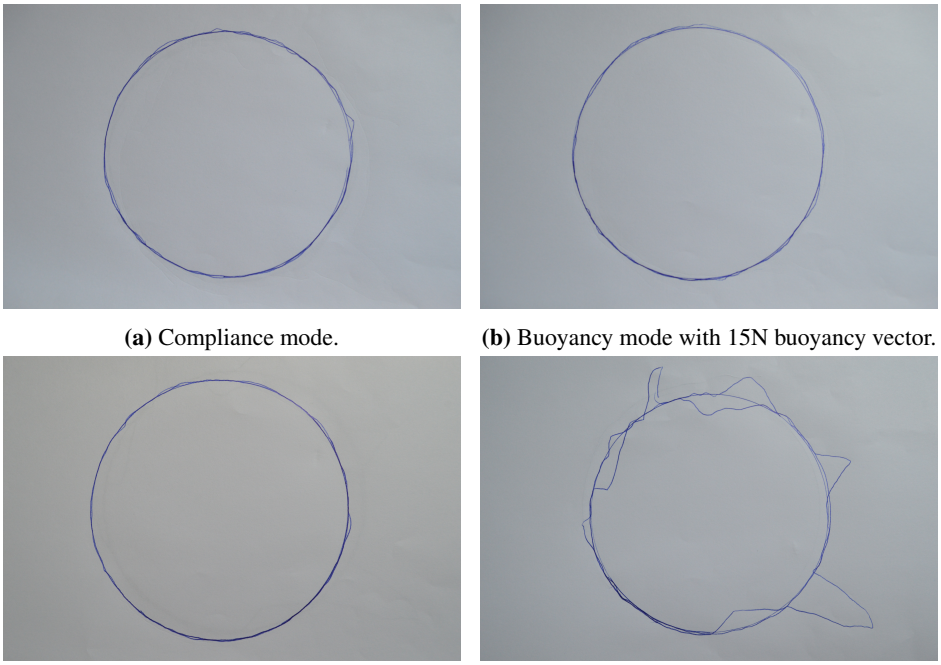
(a) Low-pass filter signal attenuation relative to the frequency of the signal. Figure taken from [4].



(b) Low-pass filter calculated group delays relative to the frequency of the signal. Without low-pass filtering enabled the F/T Net Box delivers F/T data to its Ethernet port with a delay of $286\mu s$. Figure taken from [4].

B.3 Additional experimental verification tests

Priorities and time restrictions prevented an additional experimental verification test from being presented in the main text. The additional verification test involved drawing a circle on a piece of paper aided by a reference circle. The test focused more on the visual presentation to the audience through the video recordings then practical ADL based activities. Surprisingly, the 2-plane mode struggled to perform the task due to the inability to move the wrist. Therefore, the failed tests with the 2-plane mode is substituted with the unrecorded control test results from compliance mode. The limited relevance, lack of comparable results from the 2-plane mode and time restrictions caused the test results from being omitted from in the main text. The resulting circles and associated video recordings are therefore briefly presented in this appendix without a proper analysis. Observe that the behaviour of random mode is nicely captured in the video and illustrated in figure B.5d. The results from the compliance mode, buoyancy mode[1], resistance mode[2] and random mode[3] is illustrated in figure B.5.



(a) Compliance mode.

(b) Buoyancy mode with 15N buoyancy vector.

(c) Resistance mode with 50% resistance.

(d) Random mode with 100% magnitude.

Figure B.5: Pictures of the resulting circles form the drawing test.

[1] https://youtu.be/h1lUZHUU2uQ
[2] https://youtu.be/SSMxN-hEpv0
[3] https://youtu.be/-6k2lrVp5A4

Appendix C

Safety documentation

This Appendix contains safety documentation relevant for the robotic rehabilitation system. The risk assessment performed as part of the early testing and development of the robotic rehabilitation system is presented in its entirety.

Relevant sections of the TÜV NORD certificate for the UR5 from Universal Robot is presented. Pages in the certificate not directly relevant to the UR5 is omitted from the Appendix. Please note that the testing presented in the TÜV NORD certificate is performed on the newest 3^{rd} generation internal controller with software version 3.1. Obtaining the original certificate from the 2^{nd} with software version 1.6.90722 proved to be challenging.

Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

31.03.2017

Report

Risk assessments of the UR5 robot for use in master thesis

Summary

This report gives an overview of the persons and equipment associated with the use of a UR5 robot in a master thesis during the spring of 2017. The report includes an overview of the risks and security measures connected with the use of a UR5 robot. The safety instructions and risk assessments developed by Universal Robots for the relevant firmware version of the UR5 in the department's laboratory is available here.

Inspection - risk assessment of the UR5

Dato:	31.03.2017
Present:	Mads Johan Laastad, 5th year master student

Report

Dato:	31.03.2017
Author:	Mads Johan Laastad, 5th year master student

Content

1 Introduction 2
2 Personell 2
2.1 Involved persons 2
3 Description of the UR5 system 3
3.1 Human - robot interface 3
3.2 Safety equipment 3
4 Risk assessment of the UR5 robot 4
4.1 Risk matrix 4
4.2 Security measures 6
Safety instruction for users of the UR5 robot 7
Approval of operator 7

1 Introduction

Department of Engineering Cybernetics owns a UR5 robot produced by Universal Robots AS. The applications are demonstrations of robotics for students, research and development of 6DOF robotics, etc. This report summarises the safety measures that have been implemented in order to ensure safety during use, operation and code development on the UR5 robot associated with a master thesis developed during the spring of 2017.

2 Personell

2.1 Involved persons

Person	Role
Øyvind Stavdahl	Associate professor and main supervisor.
Erik Kyrkjebø	Co-supervisor.
Mads Johan Laastad	5th year master student. Writing master thesis and using the UR5 robot for testing and developing for experimental applications.
Audience	Simple interactions during supervision of approved operator.

3 Description of the UR5 system

The UR5 system consist of:

- A 6 DOF industrial robot manipulator, consisting of six revolute joints.
- Controller cabinet containing essential firmware for control of the UR5 robot. The cabinet can be locked with a special key.

- Tablet with a GUI interface for controlling the UR5 robot. The tablet is connected to the controller cabinet through an electronic umbilical cord, and contains a physical emergency stop button.

3.1 Human - robot interface

As part of the master thesis an experimental interface between the UR5 robot and a human operator has been developed. The interface consists of a metal hoop and a wristband. An early prototype can be seen in figure below. In later prototype versions the rubber wristband has been replaced with a leather wristband for added durability.

3.2 Safety equipment

1. Physical emergency stop button
2. Several firmware safety mechanisms
3. Several new software safety mechanisms developed by 5th master student

Proper use of the UR5 robot always includes having the emergency stop button within an easy reach. There are no requirements for the UR5 robot to be operation within a physical cage, but it is encouraged to keep the working space free from any unnecessary exposure to persons or obstacles.

4 Risk assessment of the UR5 robot

4.1 Risk matrix

This chapter summarises the risks associated with the operation and use of the UR5 robot. The following risk matrix describes the operation risks prior to implementing preventive security measurements.

Risk matrix

CONSEQUENCE						
	Very serious E					
	Serious D	3	6			
	Moderate C					5,7
	Small B					
	Very small A				2	1,4
		Very small 1	Small 2	Moderate 3	Large 4	Very large 5
		PROBABILITY				

Risks - summary

Nr.	Risks
1.	Unauthorized personnel gets access to the UR5 robot
2.	Unauthorized startup of the UR5 robot system
3.	Unwanted behaviour by the UR5 robot firmware
4.	Collision between the UR5 robot and objects
5.	Collision between the UR5 robot and persons
6.	Hacking or cracking of the UR5 robot
7.	Unwanted behaviour by the experimental and newly developed UR5 robot software.

Explanation of the colors which are used in the risk matrix

Color		Description
Red		High risk. Efforts must be made to reduce the risk.
Yellow		Assessment area. Efforts should be considered based on a cost-/benefit assessment.
Green		Acceptable or negligible risk. Effort are not necessary, but can be assessed based on other considerations.

Guidelines for quantification of probability and consequences

Probability is assessed based on the following criterias:

Very small 1	Small 2	Moderate 3	Large 4	Very large 5
1 incident every 50 years or less	1 incident every 10 years or less	1 incident every year or less	1 incident every month or more	Happens weekly

Consequences is assessed based on the following criterias:

Gradeing	Human	Environment, water, earth and air	Economy/material	Reputation
E Very serious	Death	Very long term and irreversible damage.	Operations/activities stopped >1 year.	Credibility and respect substantially and lastingly weakened.
D Serious	Serious injury. Risk of permanent disability.	Long term damage. Long recovery time.	Operations stopped > ½ year. Activities stopped for up to 1 year.	Credibility and respect considerably weakened.
C Moderate	Serious injury.	Minor damage and long recovery time.	Operations/activities stopped <1 month.	Credibility and respect weakened.
B Small	Injury that requires medical treatment.	Minor damage and short recovery time.	Operations/activities stopped <1 week.	Negative effect on credibility and respect.
A Very small	Injury that requires first aid.	Insignificant damage and short recovery time.	Operations/activities stopped <1 day.	Small effect on credibility and respect.

Risk value = Probability x Consequences

Calculate risk value for humans. Independently assess if risk value calculations based on environment, economy/materiels and reputation are necessary.

4.2 Security measures

This chapter describes the security measures that have been implemented associated with each risk factor described in the risk matrix.

Risks	Security measures	Status	Risk		
			Probability	Consequence	Value
1. Unauthorized personnel gets access to the UR5 robot	• Access to laboratory limited to authorized personnel • Visual inspection of hardware damage	Preventive measures taken.	**5**	**A**	**5A**
2. Unauthorized startup of the UR5 robot system	• Access to laboratory limited to authorized personnel • Making significant firmware changes requires a password	Preventive measures taken. Password access limited, but has not been changed for some time.	**4**	**A**	**4A**
3. Unwanted behaviour by the UR5 robot firmware	• Never assuming that the UR5 firmware is completely safe • Emergency stop button	Mental preventive measures taken.	**1**	**D**	**1D**
4. Collision between the UR5 robot and objects	• Limiting objects inside the workspace • Firmware safety mechanisms • Software safety mechanisms • Emergency stop button	Preventive measures taken. Only object inside the workspace should be a small table. All other object removed.	**5**	**A**	**5A**
5. Collision between the UR5 robot and persons	• Limiting persons inside the workspace • Firmware safety mechanisms • Software safety mechanisms. Including position, velocity and acceleration limitations. • Emergency stop button	Preventive measures taken. Workspace limited to minimize clamping hazards, velocity and acceleration limited by software. Always keeping the emergency button available.	**5**	**C**	**5C**
6. Hacking or cracking of the UR5	• Limited physical access	Preventive measures taken.	**2**	**D**	**2D**

robot	• Wireless internet connection on stationary PC which require username and password upon each new session • Removing firmware cabinet key from the laboratory	No permanent ethernet connection to robot system or stationary PC on the same local network. Cabinet key moved to locker inside the operator's office.			
7. Unwanted behaviour by the experimental and newly developed UR5 robot software.	• Changes in experimental code require safety checks* before further use. • Always operating with the physical emergency stop button in the other hand.	Preventive measures taken. Safety checklist developed and implemented.	**5**	**C**	**5C**

***Safety checks**: This procedure include five mandatory safety steps that must be performed after any significant code changes. Any unwanted behaviour or deviations from expected behaviour require a review of the changes in the code and a reinitialization of the safety checklist.

1. Run new code while only observing the robot
2. Run new code and provide a simple impulse interaction with the system.
3. Run new code and move the end-effector by grasping the metal hoop lightly with two fingers
4. Run new code and move the end-effector by grasping the metal hoop firmly
5. Run new code and move the end-effector by using the metal hoop as intended, but without attaching the wristband.

Safety instruction for users of the UR5 robot

Generel

1. Use of the UR5 robot should always be authorized by key personnel inside the department of engineering cybernetics.
2. First time users should get a brief introduction by a key personnel.
3. Only operators with special insight should be allowed to change the software code.
4. The working space should contain a minimum number of obstacles and persons.
5. All use should be performed with the emergency stop button readily available.

Before starting the UR5 system

6. Minimize the number of objects and persons inside the workspace.

7. Perform a visual inspection of the hardware for any tampering or damages.

During use

8. Perform safety check if there has been made any changes to the software code regardless of the type of change that was made.
9. Always keep the emergency stop button readily available.
10. Be observant of persons entering the workspace.

Before the system is vacated

11. Make sure the robot is completely turned off before leaving the laboratory.
12. Turn off the stationary computer connected to the UR5 robots local network. Make sure the UR5 robot is not connected to the internet.
13. Enable the emergency stop button for an extra level of security.

Approval of operator:
The following is an approval for use of the UR5 robot in experimental applications as part of a master thesis during the spring of 2017. The operator should be aware of the potential dangers and comply with the safety measurements described in this risk assessment document.

Date and signature
27.04.2017
Mads Johan Laastad

Operator - Mads Johan Laastad

Date and signature
20.04.2017

Main supervisor - Øyvind Stavdahl

Date and signature
21.04.2017

Co-supervisor - Erik Kyrkjebø

ZERTIFIKAT
CERTIFICATE

Hiermit wird bescheinigt, dass die Firma / *This certifies, that the company*

Universal Robots A/S
Energivej 25
DK-5260 Odense S
Denmark

berechtigt ist, das unten genannte Produkt mit dem abgebildeten Zeichen zu kennzeichnen.
is authorized to provide the product mentioned below with the mark as illustrated.

Fertigungsstätte:
Manufacturing plant:

Universal Robots A/S
Energivej 25
DK-5260 Odense S
Denmark

Beschreibung des Produktes:
(Details s. Anlage 1)
Description of product:
(Details see Annex 1)

Universal Robots Safety System URSafety 3.1
for **UR10, UR5** *and* **UR3 robots**

Geprüft nach:
Tested in accordance with:

EN ISO 13849-1:2008, PL d

Registrier-Nr. / *Registration No.* 44 207 14097602
Prüfbericht Nr. / *Test Report No.* 3515 4327
Aktenzeichen / *File reference* 8000443298

Gültigkeit / *Validity*
von / *from* 2015-06-02
bis / *until* 2020-06-01

Zertifizierungsstelle der TÜV NORD CERT GmbH

Essen, 2015-06-02

TÜV NORD CERT GmbH Langemarckstraße 20 45141 Essen www.tuev-nord-cert.de technology@tuev-nord.de

Bitte beachten Sie auch die umseitigen Hinweise
Please also pay attention to the information stated overleaf

ANLAGE
ANNEX

zum Zertifikat Registrier-Nr. / ***to Certificate Registration No.*** **44 207 14097602**

Allgemeine Angaben: *General information:*	**The system under test and certification is the URSafety 3.1, but not the entire UR10, UR5 and UR3 robots**
Typenbezeichnung: *Type designation:*	**Universal Robots Safety System URSafety 3.1 for UR5**
Gelenk: *Joint:*	**6 rotating joints (3 x joint size 3, 3 x joint size 1)**
Nennspannung: *Nominal voltage:*	**100-240 VAC, 50-60 Hz**
I/O Spannung: *I/O power supply:*	**24 V, 2 A in control box**
Umgebungstemperatur: *Ambient temperature:*	**0-50 °C**
Nutzlast des Roboters: *Payload of the robot:*	**5 kg**
Reichweite des Roboters: *Reach of the robot:*	**850 mm**
Auslenkung der Gelenke: *Joint range:*	**±360° on all joints**
Max. Geschwindigkeit im Betrieb: *Max.speed in normal operation:*	**Joint: 180°/s** **Tool: approx. 1 m/s**
Schutzart: *Protection class:*	**Robot arm: IP 54** **Control box and teach pendant: IP 20**

F. Hegenfeld

Zertifizierungsstelle der TÜV NORD CERT GmbH — Essen, 2015-06-02

ANLAGE
ANNEX

zum Zertifikat Registrier-Nr. / ***to Certificate Registration No.*** **44 207 14097602**

Allgemeine Angaben: *General information:*	**The system under test and certification is the URSafety 3.1, but not the entire UR10, UR5 and UR3 robots**	
Typenbezeichnung: *Type designation:*	**Universal Robots Safety System URSafety 3.1 for UR5**	
Hardwareversion: *Hardware version:*	**Safety Control Board:**	**Revision B**
	Joint size 1:	**Revision H**
	Joint size 3:	**Revision I**
	Screen board:	**Revision D**
	Current distributer	**Revision F**
Softwareversion: *Software version:*	**SCB µCA:**	**3.1.453**
	SCB µCB:	**3.1.189**
	Joint 0:	**3.1.17195**
	Joint 1:	**3.1.17195**
	Joint 2:	**3.1.17195**
	Joint 3:	**3.1.17195**
	Joint 4:	**3.1.17195**
	Joint 5:	**3.1.17195**
	Teach Pendant 1, 2:	**3.1.17195**
	Polyscope	**3.1.17195**
Checksumme: *Checksum:*	**SCB µCA:**	**0x5C6D7EE1**
	SCB µCB:	**0xC447B415**
	Joint 0:	**0xC0592E78**
	Joint 1:	**0xB3D67736**
	Joint 2:	**0x90602481**
	Joint 3:	**0x052D3334**
	Joint 4:	**0xC8F438BE**
	Joint 5:	**0xC5E3E9A1**
	Teach Pendant 1, 2:	**0xBF02C153**
	Polyscope	**0x20439EA3**

F. Hegenfeld

Zertifizierungsstelle der TÜV NORD CERT GmbH — Essen, 2015-06-02

ANLAGE
ANNEX

zum Zertifikat Registrier-Nr. / ***to Certificate Registration No.*** **44 207 14097602**

Allgemeine Angaben: *General information:*	**The system under test and certification is the URSafety 3.1, but not the entire UR10, UR5 and UR3 robots**
Typenbezeichnung: *Type designation:*	**Universal Robots Safety System URSafety 3.1 for UR10, UR5 and UR3 robots**

Geprüfte Sicherheitsfunktionen:
Safety functions tested:

- **Emergency Stop (Execution monitoring of the emergency stop)**
- **Safeguard Stop (Execution monitoring of the safeguard stop)**
- **Joint Position Limit (Monitoring of the joint position limit)**
- **Joint Speed Limit (Monitoring of the joint speed limit)**
- **Joint Torque Limit (Monitoring of the joint torque limit)**
- **TCP Pose Limit (Monitoring of the TCP pose limit)**
- **TCP Speed Limit (Monitoring of the TCP speed limit)**
- **TCP Force Limit (Monitoring of the TCP force limit)**
- **Momentum Limit (Monitoring of the momentum limit)**
- **Power Limit (Monitoring of the power limit)**
- **System Emergency Stop Output (Monitoring of the System Emergency Stop Output)**
- **Robot Moving Digital Output (Monitoring of the Robot Moving Digital Output)**
- **Robot Not Stopping Digital Output (Monitoring of the Robot Not Stopping Digital Output)**
- **Reduced Mode Digital Output (Monitoring of the Reduced Mode Digital Output)**
- **Not Reduced Mode Digital Output (Monitoring of the Not Reduced Mode Digital Output)**

F. Hegenfeld

Zertifizierungsstelle der TÜV NORD CERT GmbH — Essen, 2015-06-02

ANLAGE
ANNEX

zum Zertifikat Registrier-Nr. / *to Certificate Registration No.* 44 207 14097602

Allgemeine Angaben: *General information:*	**The system under test and certification is the URSafety 3.1, but not the entire UR10, UR5 and UR3 robots**
Typenbezeichnung: *Type designation:*	**Universal Robots Safety System URSafety 3.1 for UR10, UR5 and UR3 robots**
Kategory: *Category:*	**The system possesses a heterogeneous system architecture. Detailed information about the category acc. EN ISO 13849-1:2008 for each safety function can be found in the technical report (report No. 3515 4327, 2015-05-27).**
Wichtiger Sicherheitshinweis: *Important safety notice:*	**The robot is designed to be programmed/ integrated by users for specific collaborative applications. A risk assessment is therefore required for each installation of the robot. It is particularly important that all information in the user manual should be read with close attention and all the safety instructions in the user manual must be rigorously followed.**

F. Hegenfeld

Zertifizierungsstelle der TÜV NORD CERT GmbH — Essen, 2015-06-02

TÜV NORD CERT GmbH Langemarckstraße 20 45141 Essen www.tuev-nord-cert.de technology@tuev-nord.de