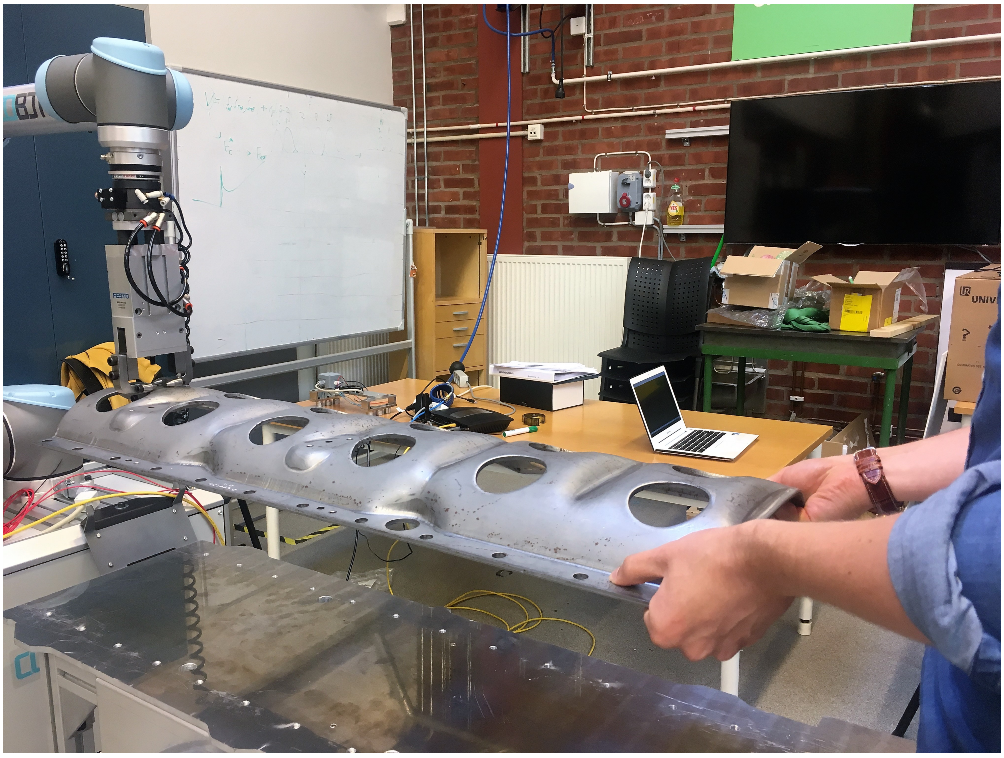
CHALMERS
UNIVERSITY OF TECHNOLOGY

Collaborative lifting between human and robot

Development of an algorithm used for mounting of a steel frame in collaboration with a human



EENX15-18-28

Authors:
Maria Blomqvist
Fredrik Hammar
Jakob Sjudin

Supervisor: Ramin Jaberzadeh Ansari
Examiner: Yiannis Karayiannidis

June 13, 2018
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden

Bachelor of science thesis 2018: EENX15-18-28

Collaborative lifting between human and robot - development of an algorithm used for mounting of a steel frame in collaboration with a human

Maria Blomqvist, Fredrik Hammar, Jakob Sjudin

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Systems and Control
Chalmers University of Technology
Gothenburg, Sweden 2018

Collaborative lifting between human and robot - development of an algorithm used for mounting of a steel frame in collaboration with a human

©Maria Blomqvist, Fredrik Hammar & Jakob Sjudin, 2018

Supervisor: Ramin Jaberzadeh Ansari
Examiner: Yiannis Karayiannidis

Bachelor of science thesis 2018: EENX-15-18-28
Department of Systems and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 10 00

Cover: Picture of collaborative lift between a human operator and a UR10 manipulator. *Photo by authors*, Gothenburg.

Sammandrag

Detta projekt innefattar utvecklingen av en algoritm för en robot, en Universal Robot 10, som i samarbete med en människa ska montera ett objekt, en stålram, på en motor. Målet är att uveckla en algoritm som gör att människan upplever lyftet som om det vore tillsammans med en annan människa. Problemet inom projektet är att få robotens rörelse så intuitiv och användarvänlig för människan som möjligt. För att manipulatorn ska kunna greppa, lyfta och placera stålramen så är någon form av verktyg nödvändig. Detta projekt utvärderar två olika typer av verktyg med målet att komma fram till vilket verktyg som är bäst lämpat för uppgiften. Det första verktyget använder sig av elektromagneter för att greppa stålramen och det andra använder sig av en pneumatiskt styrd klo. Det magnetiska verktyget bedömdes som för svagt för att pålitligt kunna hålla fast i stålramen. Det pneumatiska verktyget valdes som den mest passande lösningen på grund av dess fasta och säkra grepp.

Två olika styrmetoder presenteras för att möjliggöra förflyttning av objektet. Den första använder sig av vinkeldata för att styra och lyfta objektet och den andra använder sig av uppmätta krafter och moment vid manipulatorns ändeffektor. De två styrmetoderna utvecklades för var och ett av verktygen. Styrmetoderna testades och utvärderades utefter olika kriterier som ställs vid olika typer av tillämpningar. Båda metoder har ett antal problem som diskuteras och föreslås lösningar för. Ett huvudsakligt problem för båda styrmetoderna handlar om hur manipulatorn uppfattar människans intention endast baserat på uppmätta krafter och moment vid ändeffektorn. Styrmetoden som baseras på vinkel upplevdes som mer följsam då objektet förflyttades i det horisontella planet. Ett problem som denna metod hade var dess oscillerande beteende då det pneumatiska verktyget användes på grund av dess elastiska delar. Styrningen som baserades på kraft och moment hade inga problem med oscillering men den skapade stora moment, vilket var okompatibelt med det magnetbaserade verktyget. Det upplevdes även svårt för manipulatorn att genom endast uppmätta krafter och moment vid dess ändeffektor tolka människans avsikt och styrmetoden kunde ibland utöva krafter i oönskade riktningar.

Projektet drar slutsatsen att styrmetoden som baseras på vinkel tllsammans med det pneumatiska verktyget är den mest pålitliga och säkra lösningen. För att få denna metod och verktyg att fungera tillsammans behöver verktyget dock bli modifierat.

Abstract

This project involves developing an algorithm for a robot, a Universal Robot 10, that in collaboration with a human will mount an object, a steel frame, upon an engine. The goal is to develop an algorithm so that the human experiences the collaborative lift as if he/she were lifting together with another human. The problem within this project is to make the movement from the robot as intuitive and easy-to-use to the collaborating human as possible. In order for the manipulator to grab, lift and place the steel frame, a special tool is needed. This project evaluates two different tools in order to find the one that is most suited for the operation. The first tool uses electromagnets to grab the frame and the other one uses a pneumatic gripper to grab the frame. The magnetic tool was too weak to hold the metal frame reliably. Based on firmness of the grip and a lower risk of dropping, the frame the pneumatic tool was chosen to be the best solution.

In order to move the object, two different control methods were developed. The first one employs angle data to control and lift the object, and the second one uses measured forces and torques at the end-effector of the manipulator to dictate the movements of the frame. These methods were tested and evaluated based on different criteria and were found to have usefulness in different applications. Each method had a set of issues which are discussed and solutions are proposed. One of the main issues of both methods of control were the one related to interpreting the human's intention solely based on measured forces and torques at the end-effector of the manipulator. The angle-based controller was found to be more reliant when moving in the horizontal plane, but had issues with oscillating behaviour when paired with the pneumatic tool due to some elastic parts located on it. The force/torque-based controller had no issues with oscillation but was creating a lot of torque, which was incompatible with the magnetic tool. It was more difficult to interpret the operator's intention, with the force/torque-based controller, by only using measured forces and torques at the end-effector and would sometimes exert force in unwanted directions.

The project concluded that the solution that were most reliable and safe to the operator was the angle-based controller using the pneumatic tool. However, in order for these two to work in conjunction, the tool needed to be modified.

Acknowledgements

This report is the main documentation of a Bachelor of Science thesis at Chalmers University of Technology, Gothenburg, at the division of Systems and Control of the department of Electrical Engineering.

We would like to thank our supervisor Ramin Jaberzadeh Ansari and examiner Yiannis Karayiannidis for all the help and support throughout the project. We would also like to thank Volvo Trucks for presenting the assignment and providing the resources to complete the project.

Contents

List of Figures

Glossary

end-effector The device at the end of the manipulators arm. , 2, 5, 9–13, 15, 17–27, 29–31, 36–39, 41, 43, 44

manipulator The arms of the Universal robot 10. , 2–10, 12–21, 23–27, 29–44

pneumatics Technology that uses compressed air. , 9–11, 14, 17–21, 38, 39, 41, 43, 44

Acronyms

DOF degrees of freedom. , 3, 7, 24

ROS Robot Operating System. , 7, 11, 12, 32, 33

SFC sequential function chart. , 32

TCP tool centre point. iv, , 7, 8, 12, 14, 17–21, 23, 25, 27, 29, 30, 33–39, 43, 44

UR10 Universal Robot 10. , 6–8, 11–13, 24, 32, 33

1 Introduction

Today there is a high demand for further automation within factories and other types of production. Automation is a solution to make a production more efficient. Full automation can sometimes be difficult to implement, therefore alternatives can be looked at. A fully automated production is less flexible than having humans working in a production that can adjust to market and production changes. However, human operators can make errors because of negligence, fatigue or other factors that does not affect a machine's performance.

One way to increase the level of automation in a setting where the level is low, is to implement so called collaborative automation. Collaborative automation is a term used for describing a human and a robot working together in assembling or other types of tasks. The robots working in this kind of collaboration with humans must be classified for collaborative assembly. Within this type of automation the robot and human will work in close proximity without any safety fences. One way to perform this assembly is to have the human guide the robot throughout the operation. A benefit of this is that the human can make sure that the result is as expected [1]. The robot will normally act as a passive party compensating and/or stabilising the load. If the task is to be performed by two collaborating humans it could take time and resources for them to train in order to acquire enough skills to work with the assembly [2]. Having a robot replacing one of the humans could save time and resources.

However, collaborative automation is in itself a problem that involves making the robot and human collaborate in a natural, efficient and safe way. Mörtl et al. [3] mentions that humans can interact verbally and non-verbally, for example through gestures. It is also mentioned that the haptic interaction is challenging since it requires capabilities of interpreting the other part's behaviour on a fast timescale. Robots can use observations from human-human cooperation to calculate its own force contribution to achieve the task. The roles can however change during the execution of the task which can not be pre-determined. It can therefore be helpful to have an understanding of the physical roles within the human-robot cooperation for developing a framework for role allocation. Wojtara et al. [2] gives attention to the question of how the robot should interpret human movements and read human intentions. One problem, the most basic at the level of physical contact, is how the robot should distinguish when the human intents to rotate or translate the object. This project seeks to explore the possibilities of solving this issue as well.

This project is going to focus on developing and implementing an algorithm used for collaborative assembling of an engine. The idea originates from Volvo Trucks and a long term goal is to develop the unit until it reaches a point where it can be implemented into production. The assembly task consists of a metal frame that is to be mounted on an engine in an existing environment. The aim is to allow the human to decide the movements of the metal frame while the is robot following the human's motion.

The report has the following structure: section 2 presents the used resources within the project, section 3 describes the method and theory that has been applied in the development of the project, section 4 presents the results of testing the produced solutions and discusses them. At last, section 5 concludes the project and gives suggestions on further development.

1.1 Related work

A lot of research has been done on the topic of manipulating a jointly-held object. One important question that arises is how the manipulator is to interpret the collaborating human's movement using only force and torque sensors attached at its wrist. Ygit et al. [4] suggest that two principal ways of solving this is by either having the manipulator act fully reactively to sensor inputs given when the collaborating human exerts forces and torques upon the manipulators end-effector, or by trying to interpret the human's intentions based on previous movement. The authors recognise that one problem with these methods is that it is difficult to differentiate between torques and forces at the end-effector, meaning that it is problematic to decide if the collaborating human means to rotate or translate the object.

Dumora et al. [5] further supports the notion that it is difficult to determine whenever the human means to rotate or translate the object based on wrenches only and also suggest that the intention that be decoded by using haptic cues to classify between different types of movement, statistical analysis is used to find a correlation between haptic measures and intentions of motion. It is worth noting that the produced results are only tested in a horizontal plane which is not directly applicable to the task this project seeks to solve.

Wojtara et al. [2] discusses different methods for solving this problem by using switched or switchless algorithms. A switched algorithm requires some form of input to switch between two modes for rotation and translation of the object. One benefit of a switched algorithm is that it is decisive and movements can not be misinterpreted by the manipulator. The downside is the increased complexity from the collaborating human's perspective and some training might be needed before the operation can be executed smoothly. A switchless algorithm is similar to a switched algorithm but it uses more sophisticated means of determining whenever the human wishes to translate or rotate the object that does not require a switch. This method, if implemented correctly, may operate more similar to a human and therefore may not require as much training for the collaborative human. This, however, will mean that it is easier for the manipulator to misinterpret the intention of the human, which could lead to the lift operation failing in some way. This sets a higher demand on the implementation of the lifting algorithm and a more sophisticated way of determining what the collaborating human's intention is needed.

Another approach is to set up a different framework for manipulation of the object, Tabuko et al. [6] uses a nonholonomic constraint to only allow the object being manipulated to move similar to a wheelbarrow in the horizontal xy-plane. This means that the human can only manipulate the jointly-held object by guiding it along a path that is tangential to the defined virtual wheel. The benefit of using this method is that it is intuitive to the operator since using a wheelbarrow is something familiar to most people. This method can also be extended so that the object can be manipulated with six degrees of freedom (DOF) in a thee dimensional space.

1.2 Purpose

The purpose of this project is to develop a system where a robot and a human work together in close proximity. The system will contain an algorithm for controlling the robot along with the required hardware. The task is to mount a steel frame upon an engine by having a robot and a human work collaboratively. In this project an algorithm will be developed that makes the robot follow human motion. The aim is to obtain a motion from the robot to the human that will make the lift assignment easier than if the human would lift alone. The project is also set out to investigate how viable it is to implement collaborative assembling for a task like the one stated above. Another aim for the project is to analyse the complexity of implementing such an application and the solution will be evaluated experimentally. The possibilities of further development after the conclusion of this project will be explored.

In short, the project aims to:

- Develop an algorithm, along with the required hardware for a collaboration of a human and a robot to perform an assembly task
- Explore the possibility of facilitating an assembly task with the help of collaboration between human and robot
- Analyse the complexity in creating such an application and exploring the possibilities of further development

1.3 Problem description

The problem faced within this project was to develop and implement an algorithm for a manipulator to lift and place a steel frame upon an engine in collaboration with a human, see figure 1.1. The steel frame weights 13kg.

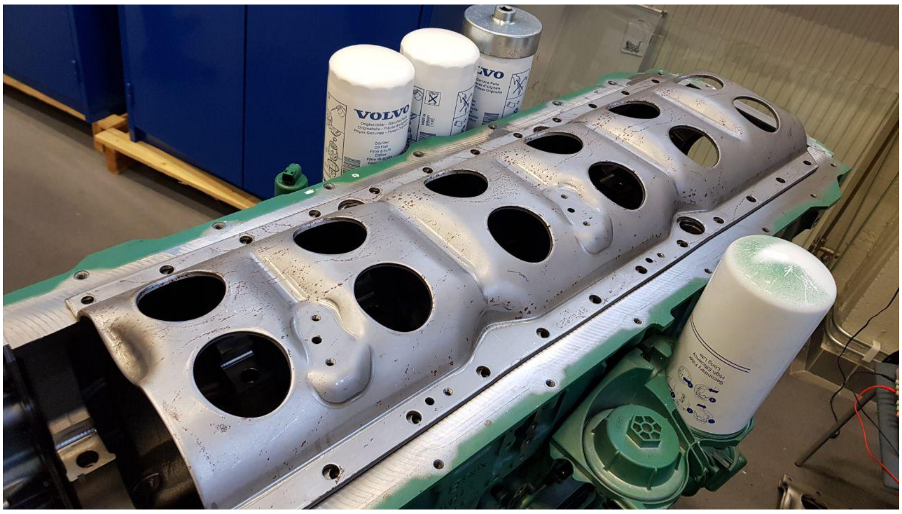


Figure 1.1: Image of the steel frame that is to be lifted and placed upon an engine.

The human will grab onto one short end of the frame and the robot on the other and then move it in collaboration. The assembly scenario is described in figure 1.2.

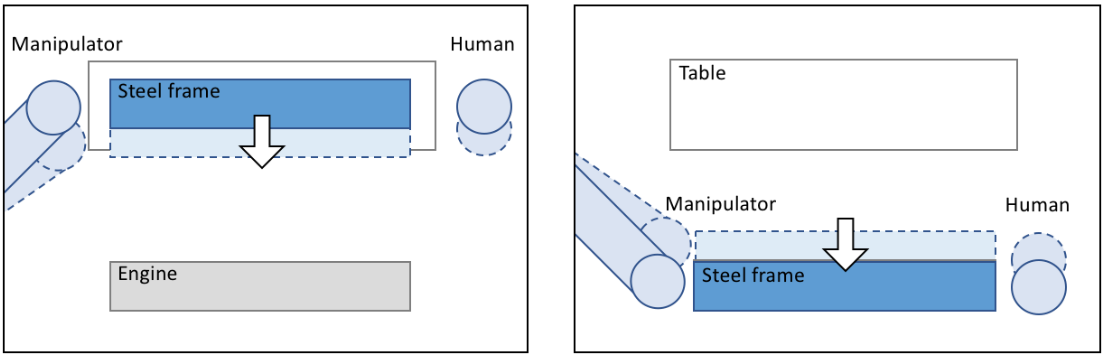


Figure 1.2: Image of the beginning of the assembly task (square to the left) and the end (square to the right).

The left image shows the beginning of the assignment and the right image shows the end. Within the assembly task the starting position of the steel frame is on a table, on a stack of steel frames or something similar. In figure 1.2, this position is represented by a table. The end position of the steel frame within the assembly task will be upon an engine. A challenge within the assembly task is to develop an algorithm that makes the robot interpret the humans intention. The mounting will consist of different sub tasks that are described further in *Method*. The problem description has been narrowed down to concrete goals that are to be achieved.

1.3.1 Goals

For the robot to be able to grab, lift and place the steel frame there is a need for some sort of specialised end-effector in place that allows the robot to grab the frame, hold on to it while the frame is being moved and then release the frame. This specialised end-effector will henceforth be referenced to as the *tool*. The tool will be mounted on the end-effector of the robot and therefore has to be compatible with the manipulator. Since the application for this tool is rather specific to this project, the aim will be to develop and test a tool that can handle the requirements needed for implementation. The goals for the lifting tool are to design a tool that can grab, lift and place the steel frame and to develop software for it. Further, there is a requirement of modularity of the tool. Since there are several operations in an assembly task, the tool must be designed in such a way that it is changeable on demand. Moreover, the weight of the tool is of importance since the payload of the manipulator is limited and therefore a low weight of the tool is desirable.

The manipulator has the assignment to move the steel frame in a path guided by the collaborating human. Data will be gathered by the manipulator by reading different values from sensors including force, torque etc. The goals for the manipulator is to find an appropriate way to collect and process the measured data and to implement algorithms to enable it following human movement. A goal is to make the human be able to show the robot in an easy way in which direction he/she wants the robot to move the steel frame. In order to perform the task, the manipulator and the tool need a way to communicate with each other. One example is that the tool needs to "know" when the task is beginning and it should grab the steel frame and when the task is ending and it should release it.

In short, the goals are to:

- Manufacture a tool for the manipulator to grab and handle the steel frame
- Evaluate the tool to check the rigidity of the grasp and clarity of the measured data
- Develop an algorithm for the manipulator to handle the object cooperatively with the operator
- Enable the communication between the manipulator and the tool
- Evaluate the safety of the tool, the manipulator and the operation

- Coordinate the manipulator and the gripping tool in the beginning and the end of the assignment

1.4 Boundaries

This project is restricted to producing an algorithm for solving the task for a specific type of robot - Universal Robot 10. The solution is fitted for this robot and in order to apply it using other types of robots it might need some sort of alterations. The object that is dealt with within the assignment is a specific item - the steel frame. Therefore, the solution is not intended to grab, lift and place other objects that differ from the steel frame regarding shape, weight and material. The task for the robot is specific and therefore the solution will be produced with the goal to solve only this particular task. The solution will be fitted for the robot's current working environment and for example not take into account cords that are laying on the floor or other possible disturbances.

1.5 Questions to answer

After achieving the set goals there are a number of questions expected to be answered in order to estimate whether the produced solution can be seen as successful or not. These questions are as follows:

- What risks concerning the human safety are involved in the produced solution?
- Does the solution feel intuitive to an inexperienced operator when compared to human-human collaborative lifting?
- How can the produced solution be developed further?

The questions will be answered by performing physical tests on the produced solution together with analytic discussion. These answers are presented throughout the report.

2 Resources

In this section, the resources that have been used within the project will be presented and described further.

2.1 Manipulator

The manipulator used for moving the steel frame is a Universal Robot 10 (UR10). The Robot Operating System (ROS) which the Universal Robot 10 uses is called Polyscope [7]. The six rotating joints it consists of are called shoulder, elbow, wrist 1, wrist 2 and wrist 3 (see figure 2.1).

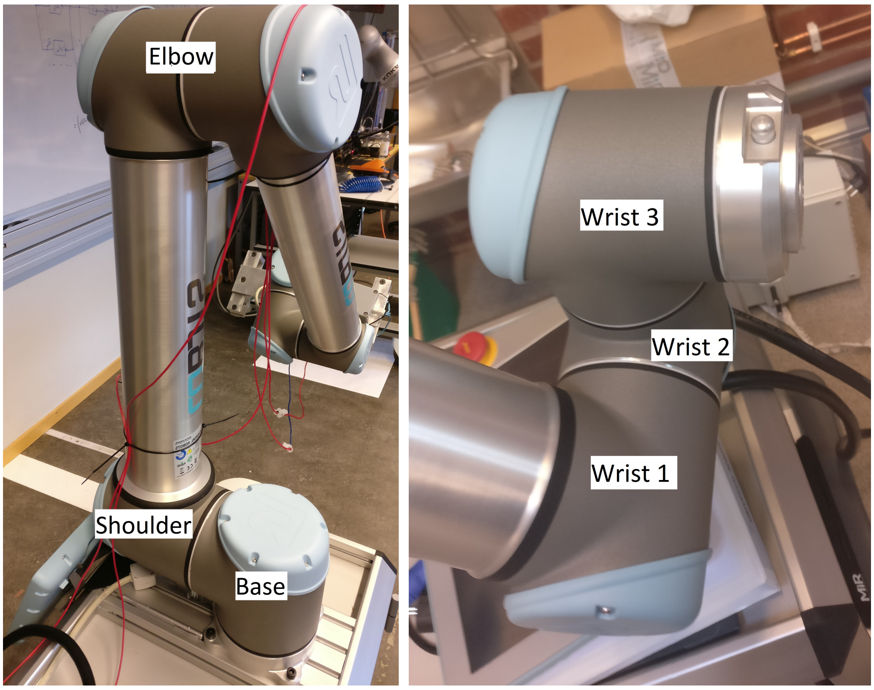


Figure 2.1: The manipulator and the joints it consists of.

These wrists enables the manipulator to have six DOF. These DOF are x, y, z, r_x, r_y and r_z. The x, y and z coordinates are oriented with the base joint as the origin point, henceforth called base-space (see figure 2.2). r_x, r_y and r_z represent rotation around their respective axes. Another coordinate system that is of importance is the space originating from the tool centre point (TCP). The TCP-space is the point that is usually defined by the contact point the manipulator has with an object. In the case of lifting collaboratively with a human, this point is defined as the point where the tool grips the object.

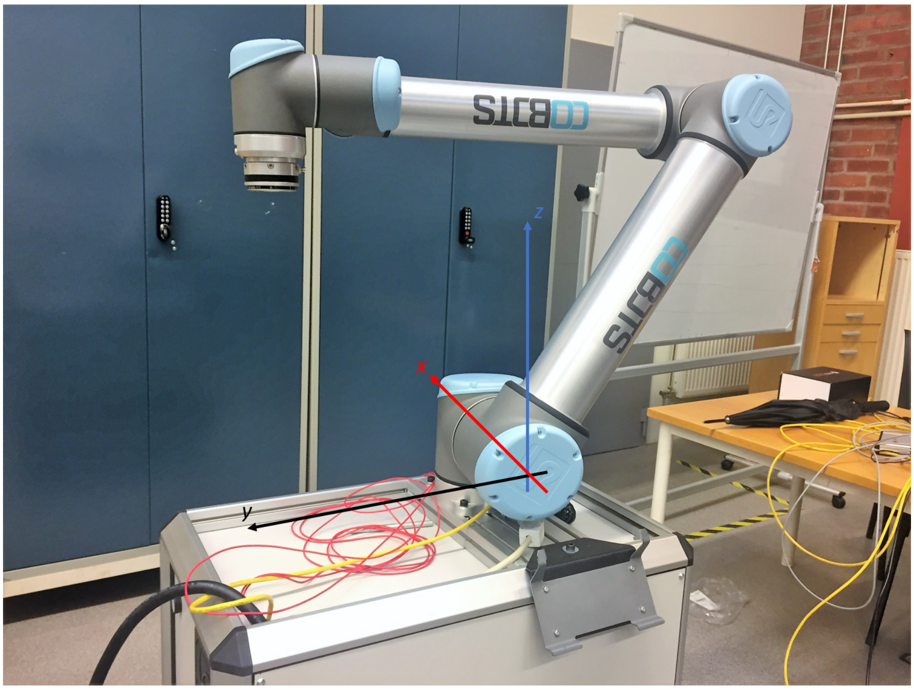


Figure 2.2: Image of the base coordinate-system of the manipulator.

The UR10 is controlled either by defining points $\begin{bmatrix} x & y & z & r_x & r_y & r_z \end{bmatrix}^\intercal$ in the base-space or TCP-space that the manipulator will move to relative the coordinate-system of choice, or by sending velocity commands. Velocity commands consist of a vector $\bar{v} = \begin{bmatrix} v_x & v_y & v_z & \omega_x & \omega_y & \omega_z \end{bmatrix}^\intercal$, an acceleration a_max and a variable min_time. The vector $\bar{v}$ denotes the velocities of each of the points in the TCP-space, a_max is the maximal acceleration permitted and min_time is the minimal amount of time that the command will be executed before returning [8].

A restriction of the manipulator is that it can handle a recommended maximum payload of 10kg with a reach of 1300mm [8]. It is possible for the manipulator to exert forces of up to 250N but it runs the increasing risk of protective stops and joint slipping with the increased force usage.

The manipulator is mounted on an automated guided vehicle, which would allow it to move within its environment. However, this will not be used within this project.

2.1.1 Teach pendant and Polyscope

The teach pendant is a built in computer with a touch-screen, emergency-stop and power button connected to the manipulator. It is the original method for using the robot and does not require any external methods for controlling the manipulator. The teach pendant runs on the Polyscope operating system which is a graphical interface created for the Universal Robot series. Therefore, it contains all the intended functions such as installing robot software, adjusting the safety restrictions, building programs and controlling the manipulator.

2.2 Gripping tools

Gripping tools are connected to the end-effector of the manipulator and this is needed in order to grab the metal frame. In the project two different tools were provided by Volvo Trucks. In the beginning of the project a magnetic tool was presented and later on in the project a pneumatic tool was introduced.

2.2.1 Magnetic tool

At the start of the project there was an already existing tool, depicted in figure 2.3, developed by Volvo Trucks for this specific task. In this image the surface facing up is the surface that is in contact with the steel frame, when the tool is connected to it. This tool has got a total of four electromagnets located on both sides of the tool and each magnet needs the voltage 24V and the current 0.13A. The magnets are attached in rails and can be moved so that the distance between the two pairs of magnets increases or decreases. The magnets are connected in series and the total weight of the tool is approximately 1.50kg.

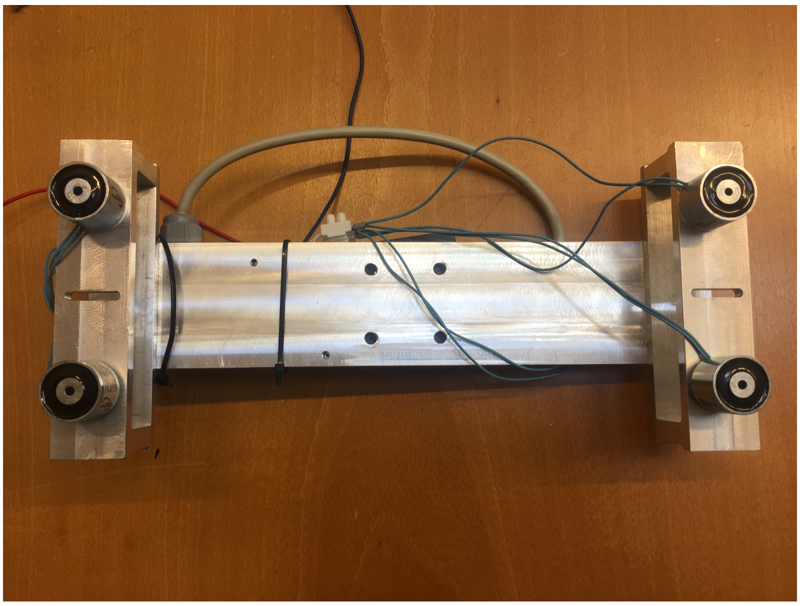


Figure 2.3: The magnetic tool with four electromagnets attached in rails.

2.2.2 Pneumatic tool

Another tool was provided by Volvo Trucks that uses a pneumatic actuator, see figure 2.4. This tool is heavier than the magnetic tool, 3.20kg, and therefore increases the payload on the manipulator when mounted on it.

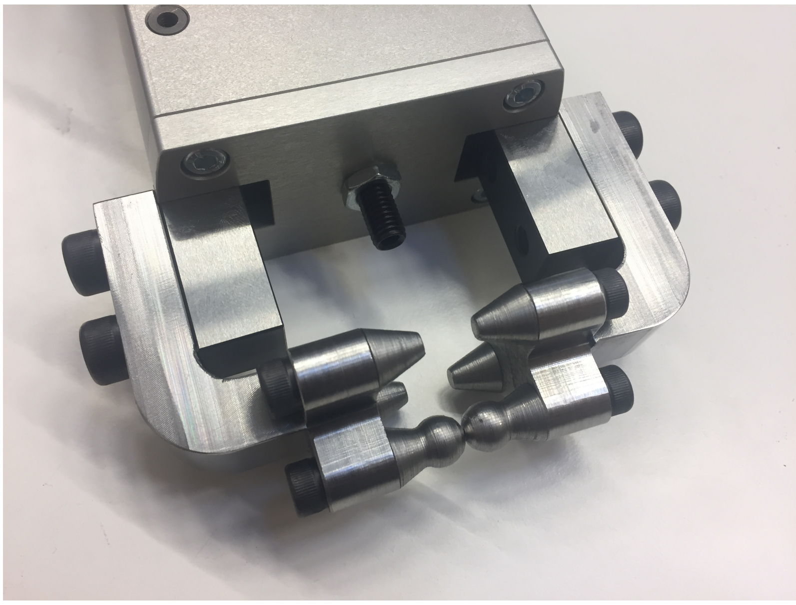


Figure 2.4: The pneumatic tool zoomed in on the contraption part of the tool.

In order to attach and detach the tool to and from the outer wrist of the manipulator there is a pneumatic interface. The outer part of the tool that grabs the steel frame is a gripper that also needs compressed air in order to open and close. Meaning that a 3-state pneumatic switch is needed to be able to open and close the gripper and also attach and detach the tool from the end-effector. These states are depicted in figure 2.5. Note that the gripper closes automatically when the tool is detached.

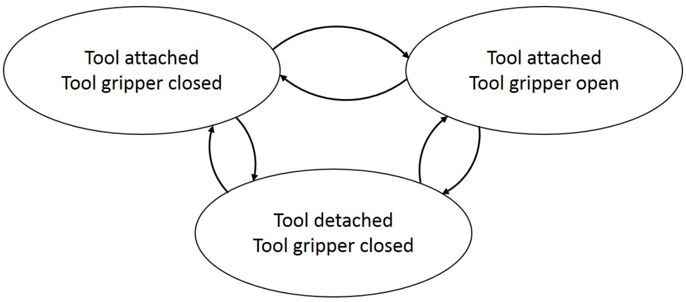


Figure 2.5: Graphed states that the tool can be present in.

One noted feature of the pneumatic tool are the rubber spacers that are positioned close to the attachment interface of the end-effector (depicted in figure 2.6). These rubber spacers add a dampening factor when the human collaborator tries to exert forces and torques upon the end-effector.

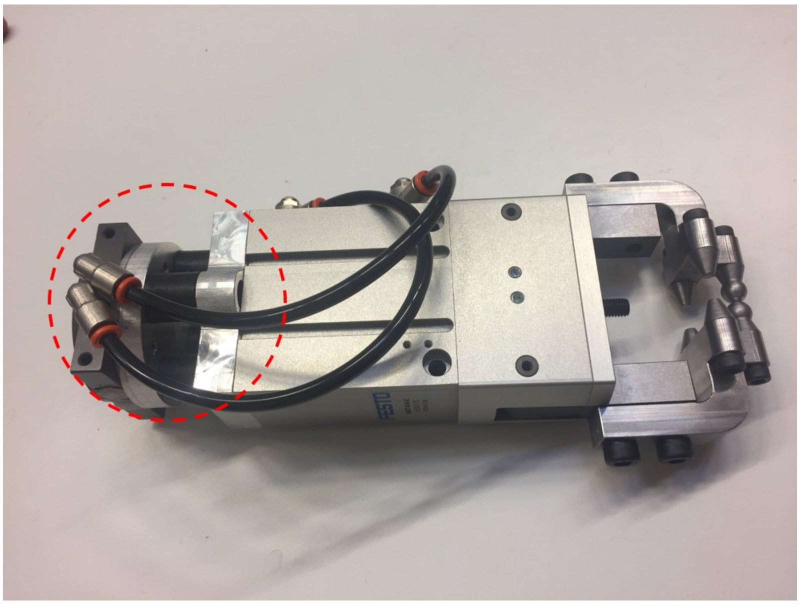


Figure 2.6: Rubber spacers positioned close to the end-effector of the manipulator.

It is worth noting that there is risk of impingement when the pneumatic tool grips the frame, when implementing a controller. It might be of interest to make sure that when the tool grips the frame, the operator does not need to keep his/her hands close to the point of contact with the frame.

2.3 Robot Operating System (ROS)

Robot Operating System, in short ROS, is a bulletin board-based operating system used for communication between different components in a system. This is done by subscribing and posting to different *topics* that different parts of the program can read and write to. It uses the Linux platform and the version of ROS used in this project is Indigo. Indigo is used because it is the latest stable release that supports the *UR modern driver* package that is used for communication with the UR10. In this project ROS is mainly used to evaluate data from different sensors and send commands to the manipulator based on this data. ROS supports Python3 and C++ among other programming languages. This project has mostly used Python3 because of the easier-to-understand syntax compared to C++ [9]. Linux is run through a virtual machine on a Windows computer.

2.3.1 Important packages

ROS is a package based operating system and all of the packages used in this project are open-source and have been downloaded from Github.com. The most important package is *UR modern driver* [10], which has been developed by Thomas Andersen [11]. This package is a driver that sets up an interface for communication with the UR10 robot through ROS. It also provides transforms from different frames of the manipulator, the most important one, in this project being the TCP-space. *UR modern driver* is updated sporadically and therefore there are some issues that has to be taken into account when using the package. Socket programming has been used to work around the cases where *UR modern driver* did not have an implemented function, an example being loading and running programs locally on the manipulator from an external device. Another important package is the Optoforce ROS package that the external force sensor attached to the end-effector uses. This package is a simple driver that returns forces and torques exerted upon the sensor as data [12].

3 Method

The development of the system for collaborative lifting between a human and a robot was split down into different sub tasks that needed to be solved separately and then integrated into the overall solution. The main focus of the project is the development of the algorithm for lifting and moving the steel frame by having a human and a UR10 working collaboratively.

Implementing the algorithm that can interpret and follow human movements only solves part of the overall problem, since there are a lot of other small operations involved in a complete mount that also need to be considered. These smaller operations might seem insignificant but they need to be implemented and working correctly in order for the system to work as a whole. These operations are depicted in a sequential function chart from the perspective of the manipulator, see figure 3.1. This project focuses mainly on the collaborative lifting and has solved the other tasks in figure 3.1 by setting the manipulator in *freedrive*. Freedrive is a state of the Universal Robot 10 where the manipulator is completely compliant and is not executing any commands, which is a form of admittance control, which includes some inertia to prevent the manipulator from moving too much. This state is used so that the collaborating human can locate and move the end-effector manually to the object so that the lifting operation can start.

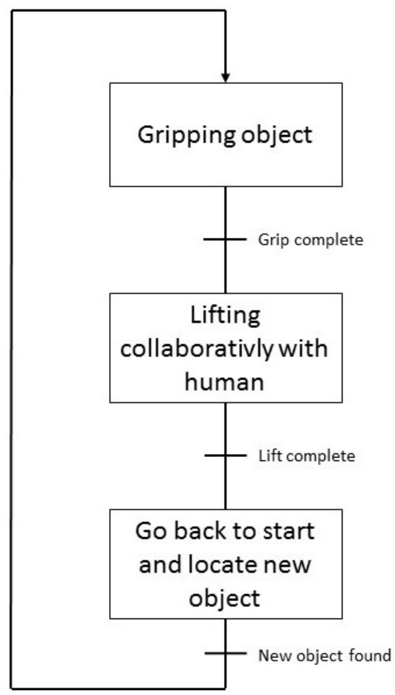


Figure 3.1: Sequential function chart of the complete lifting and placing operation.

In these smaller operations there are many other problems to consider. Figure 3.1 gives an overall view of the operation that this project seeks to implement. In order to make the system viable for use in a factory environment, it also has to be robust and fast enough so that it does not compromise the speed or safety of the workspace it is implemented into. Therefore, these smaller operations are needed to be implemented and optimised.

A couple of viable options to implement these methods of control is to use multiple sensors like camera, accelerometers or other sensors attached to the collaborating human as complement for reading and determining the movements of the manipulator. The issue that arises is the increasing complexity for the collaborating human and also the flexibility of the solution itself. Since the end goal is to have a solution that can be implemented into a specific production environment it is of importance that it is easy to use and that it does not require large modifications to the production-line already in place.

3.1 Gripping tool

The requirements of this tool is that it is reliable, as in it will not drop the frame, easy to use and safe, more specific details regarding design of the tool are listed below under *Designing the tool.* As described in *Problem description*, the main task in this project is to lift a metal frame. In order to lift the frame, a gripping tool is needed which will be able to pick up the metal frame without being in the way of the assembling onto the engine. Two tools have been provided from Volvo Trucks during the project. At the start of the project an electromagnetic-based tool was provided, later on in the project a second tool was provided, that instead of magnets used a pneumatically-driven claw that grips the metal frame.

3.1.1 Designing the tool

For the assembly task, the tool needed some specifications regarding its capacity. The following criteria were used when evaluating the two tools.

- Handling of torques along axes x, y and z:

  In the process of the collaborating human manipulating the metal frame, the manipulator will need to withstand torques along axes x, y and z in the space of the TCP. This is mainly due to the fact that the human will control the manipulator by exerting forces and torques upon it. There is also a possibility that the manipulator will need time to react to the torques exerted by the human and therefore the tool needs to manage this as well.

- Manage to hold the same share of the steel frame's weight as the manipulator is holding

During the lifting operation the weight of the steel frame will be divided between the manipulator and the human. Because of this, the tool needs to be able to hold the same share of weight that the manipulator is holding during the task.

- The fixture needs to be compatible with the manipulator:

  This criteria is needed in order to easily be able to mount and dismount the tool on the manipulator's end-effector.

- Not restraining the manipulator's performance:

  Since the manipulator only supports a recommended payload of 10kg, it is important that the tool is relatively light and does not in any other way compromise the functionality of the manipulator.

3.1.2 Evaluation of the magnetic tool

Initially, the magnets on the magnetic tool were connected in series where the current through each of the components is the same and the voltage across the circuit is the sum of the voltages across each of the magnets. This was changed into a connection in parallel where voltage across each of the magnets are the same and the total current through the circuit is the sum of the currents through each component.

The magnets connected in series and parallel circuit are shown in figure 3.2.

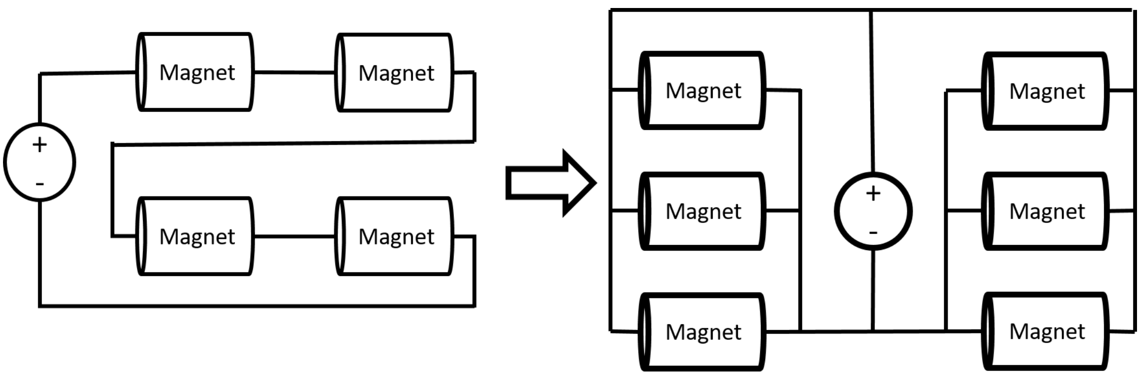


Figure 3.2: The circuit before modification (to the left) and the circuit after (to the right).

The reason that the magnets were connected in parallel is supported by Ampere's law [13], that is given as:

$$\int J \cdot dA = \oint H \cdot dl$$

where J is the current, A is the cross sectional area of the core, H is the magnetising field and l is the total length of the magnetic field path. Ampere's law says that the integral of the magnetising field H around any close loop is equal to the sum of the current J flowing through the loop. Consequently, magnets in a parallel circuit will have a larger amount of total current through the circuit than a series circuit, which leads to a stronger magnetising field H. This will result in a stronger magnetic tool compared to when the magnets were connected in series.

First, the tool was connected to a 24V power source, external to the manipulator, and it was tested. It was noted that the tool had difficulty in supporting the mass of the frame, especially when affected by external torques. There were also some problem with the magnets not getting a proper connection with the metal frame. In order to increase the power of the tool, two additional magnets were attached, which is depicted in figure 3.3. The tool became less sensitive to torque and the case that the magnets not connecting properly.

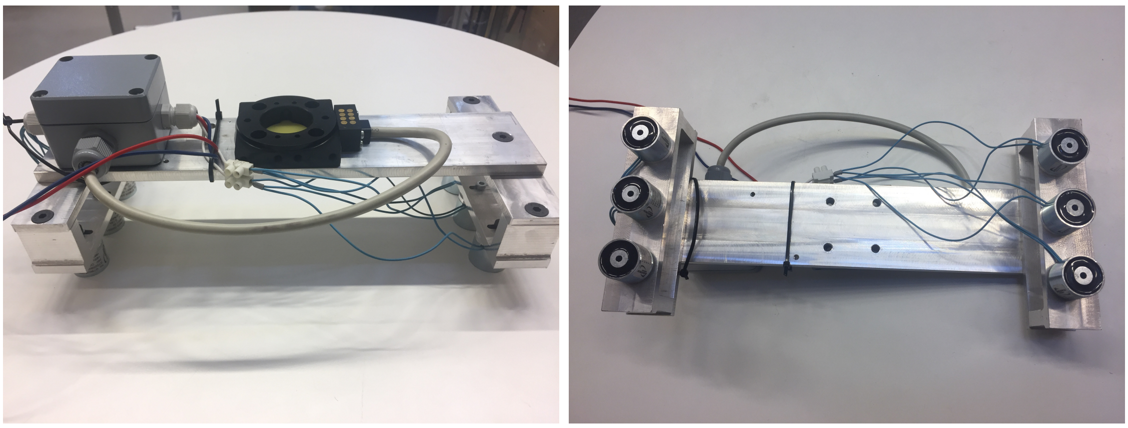


Figure 3.3: Magnetic tool seen from the top (to the left) and seen from the bottom (to the right).

However, two new problems arose: the magnets seem to be slightly different in height which resulted in the tool slightly pivoting around the middle magnets and the current increasing to almost 1A. The tool slot on the manipulator can only provide 0.6A. By connecting the tools circuit directly to the manipulator's control box which can provide 1A through a controllable digital output, the cables however had to run along the manipulator arm. The tool still had a problem with dropping the metal frame if any form of jerky motion was exhibited from the manipulator, a problem which was less common whilst connecting the tool to a external power supply rather than to the manipulator's control box. Upon further investigation whilst measuring the current in the circuit it was found to be 0.72A when connected to the control box, however the magnets are suppose to use $6 \cdot 0.13\text{A} = 0.78\text{A}$. When connected to a external power supply the current was measured to 0.77A.

3.1.3 Evaluation of the pneumatic tool

When attaching the pneumatic tool to the end-effector of the manipulator, see figure 3.4, the TCP needs to be offset along its z-axis due to the length of the tool. This is done with a homogeneous transformation matrix where $\bar{x}_{tcp}$, $\bar{y}_{tcp}$, $\bar{z}_{tcp}$ are the updated coordinates after the offset:

$$\begin{bmatrix} \bar{x}_{tcp} \\ \bar{y}_{tcp} \\ \bar{z}_{tcp} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{tcp} \\ y_{tcp} \\ z_{tcp} \\ 1 \end{bmatrix}$$

Note that the offsets $p_x = p_y = 0$ in this case, since offsetting the TCP along only the z-axis is of interest. The offset p_z is measured in millimetres and is set to 300mm.

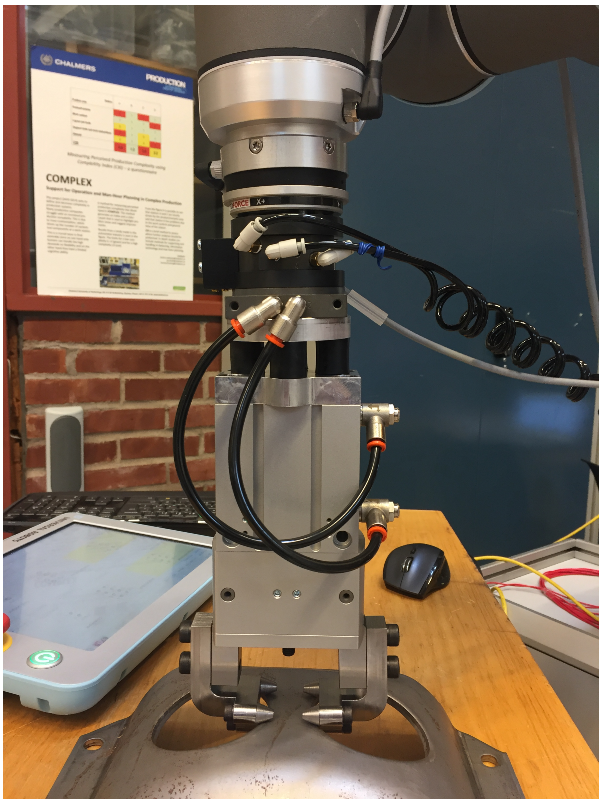


Figure 3.4: Pneumatic tool attached to the manipulator and the frame.

The pneumatic tool runs a lower risk of dropping the frame when compared to the magnetic tool, this due to the firmer grip it has on the frame when grabbing it. It is also more secure from a safety perspective since it has the option to be configured so that it will not drop the frame if power is cut to the pneumatic relay in case of an emergency stop or similar situation, compared to the magnetic tool that will drop the frame if the power is cut to it.

When mounting the pneumatic tool onto the end-effector of the manipulator, it was noticed that there was an angle offset θ_{offset} between the coordinate system of the TCP-space (x_{tcp} and y_{tcp}) and the defined coordinate system of the pneumatic tool (x_{tool} and y_{tool}), showed in figure 3.5.

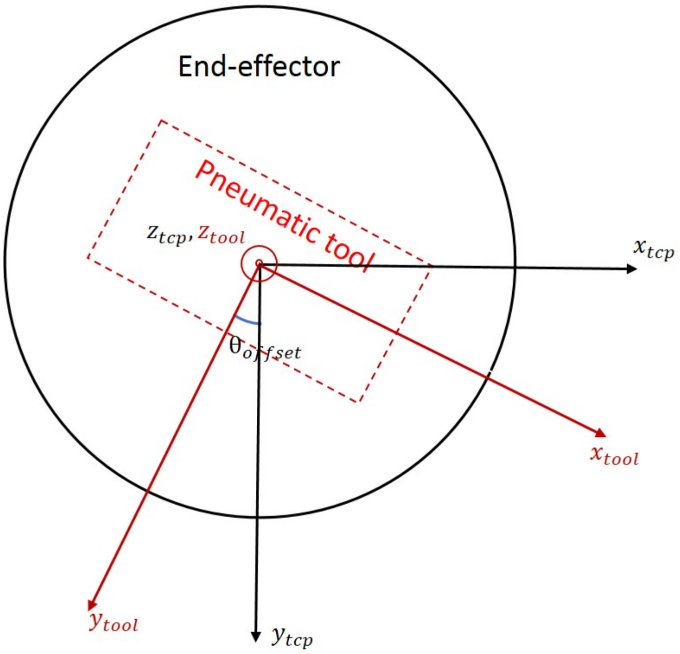


Figure 3.5: Offset angle θ_{offset} between the end-effector and the pneumatic tool seen from above.

To set up a working interface between the end-effector and the tool, the TCP-space and the space of the tool must be aligned so that:

$$\begin{bmatrix} x_{tcp} & y_{tcp} & z_{tcp} \end{bmatrix} = \begin{bmatrix} x_{tool} & y_{tool} & z_{tool} \end{bmatrix}$$

Since there is the option to redefine the TCP-space while developing the algorithm, the choice was made to align the TCP-space with the tool-space and not vice-versa. A rotation matrix was used to align the x- and y-axis of the coordinate systems of the end-effector and the tool. The rotation matrix is defined as:

$$\begin{bmatrix} \bar{x}_{tcp} \\ \bar{y}_{tcp} \\ \bar{z}_{tcp} \end{bmatrix} = \begin{bmatrix} \cos\theta_{offset} & -\sin\theta_{offset} & 0 \\ \sin\theta_{offset} & \cos\theta_{offset} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{tcp} \\ y_{tcp} \\ z_{tcp} \end{bmatrix}$$

where $\bar{x}_{tcp}$, $\bar{y}_{tcp}$ and $\bar{x}_{tcp}$ denote the TCP's coordinates axes that are aligned with the tool's coordinate axes. Note that $z_{tcp} = z_{tool}$, therefore no alignment in the z direction is needed. The θ_{offset} was measured to approximately $-70°$.

It is worth noting that the pneumatic tool is heavier than the magnetic tool and will therefore add more to the payload of the manipulator, meaning that the amount of force that the manipulator can exert upon the frame reduces in comparison to the magnetic tool.

The rubber spacers that were attached to the tool proved to cause oscillating movement from the end-effector in some situations. Hence it might be desirable to remove these spacers depending on the application of the tool. This removal, however, is not possible without making larger modifications to the tool and therefore it has not been done during the course of the project.

3.1.4 Testing tool grip security

In order to test and verify the firmness of the grip that the two different tools have on the object, two different tests were done. The first test is conducted by attaching the tool in the middle of the steel frame and applying a varying increasing force in the positive z-direction of the TCP. This varying force increases until the force sensors located at the end-effector register a value of 250N, so that there is no risk that damage will be done to the manipulator. This is done to evaluate the grip strength when the tool is being affected by forces in linear directions. The results are presented in figure 3.6.

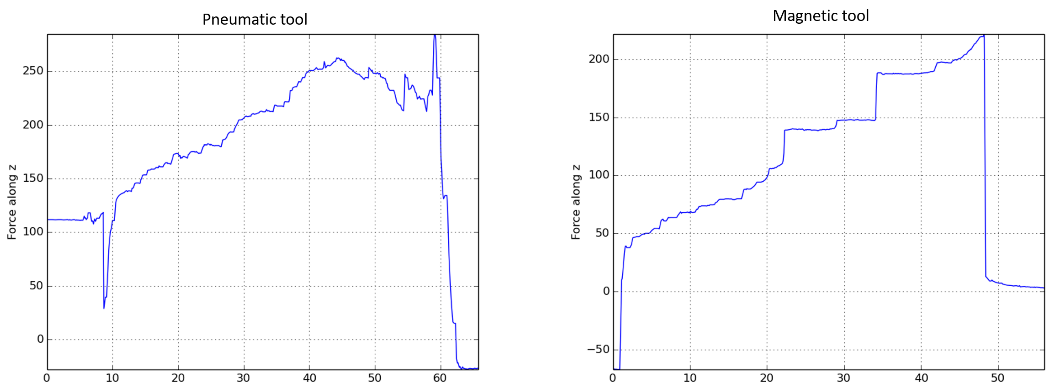


Figure 3.6: Results of a force test where on the graphs vertical-axis shows the force [N], which linearly increase along the z-axis of the TCP, and the horizontal-axis is time [s].

The pneumatic tool (image to the left in figure 3.6) could safely grasp the frame up until the 250N limit of the test was reached around 40s after the test was started. The impulse registered above 250N at approximately 60s is likely due to the sudden reset of the manipulator in order to stop exerting any force. In the rightmost image, the magnetic tool was evaluated. This tool prematurely released from the frame during the test at an applied force of approximately 220N.

The second test was designed to test the tool's grasp upon the object when torque was applied. The tools were attached in the middle of the frame and then an increasing torque was applied around the x-axis of the TCP-space. Figure 3.7 depicts two different iterations of the test where the magnetic tool would release from the frame at different torques. This shows that the tool can be somewhat unreliable regarding when it might release from the frame due to torques being exerted upon it.

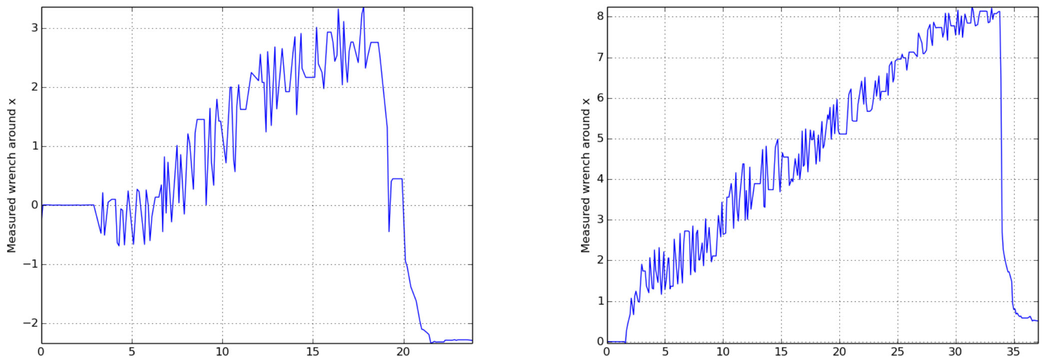


Figure 3.7: Results of a torque test conducted on the magnetic tool where on the vertical-axis of the graph is the increasing torque [Nm] applied along the y-axis of the TCP and on the graphs horizontal-axis is time [s].

Figure 3.8 shows the results from the same test being conducted on the pneumatic tool. This test was ended when the torque at the end-effector was measured to 12Nm to risk not damaging the manipulator and the external force sensor.

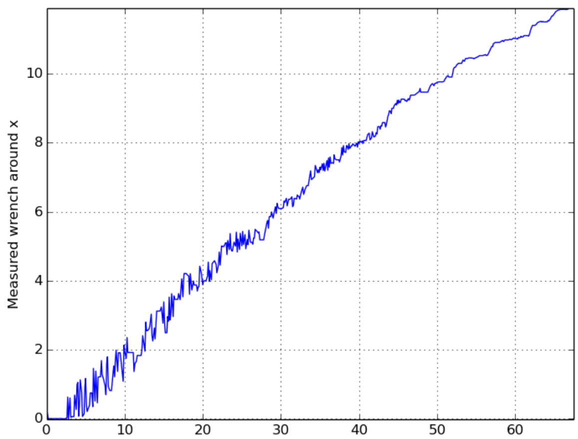
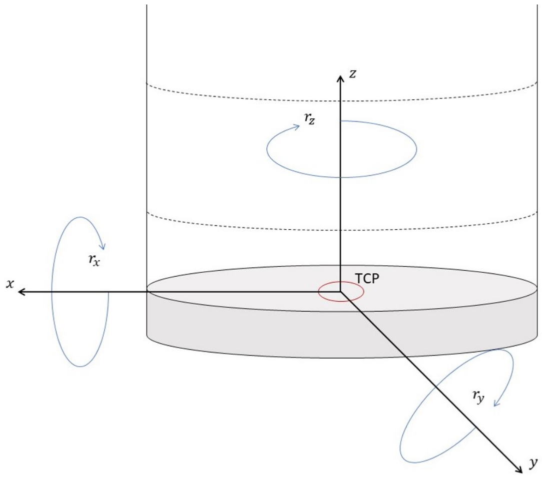


Figure 3.8: Results of a torque test conducted on the pneumatic tool where on the vertical-axis of the graph is the increasing torque [Nm] applied along the x-axis of the TCP and on the graphs horizontal-axis is time [s].

Based on these tests it can be concluded that the pneumatic tool has a firmer and safer grasp upon the object when being exposed to forces and torques. It is also worth noting that the pneumatic tool did not show any signs of releasing from the object during the tests, while the magnetic tool released prematurely during all of the tests. It can also be seen that the magnetic tool has some issues in reliability since it can be hard for the operator to verify if the tool has been properly attached to the object or not, this is exemplified in figure 3.7. Because of these results, the pneumatic tool was chosen to be the tool that would be used in the implementation. This mainly due to that it can achieve a firmer and more safe grip so that the operator does not need to worry that the manipulator will drop the frame during the collaborative lift.

3.2 Control design

A common solution for external manipulation of a robot is to use different types of admittance or impedance controllers [14][15][16]. In this project, admittance control is implemented by using measurements of forces and torques exerted upon the end-effector of the manipulator and converting those to velocities by using scaling factors. The orientation coordinate system of the end-effector, also called TCP-space, is shown in figure 3.9. Note that the origin point for the TCP-space has been redefined when the pneumatic tool is used (see *Evaluation of the pneumatic tool*).



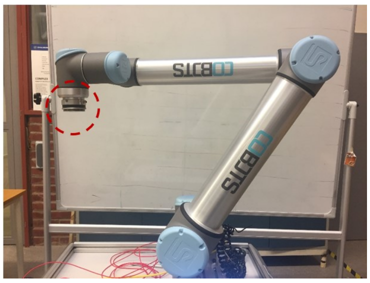


Figure 3.9: End-effector coordinate system with real manipulator for reference.

There are many different ways of using admittance control, one common problem is that it is difficult to decode the intention of the human by only reading forces and torques at the end-effector.For example, when the human moves the steel frame, it is difficult to determine whether he/she means to rotate or translate the object. This movement could for example be in the negative z-direction, see figure 3.10.

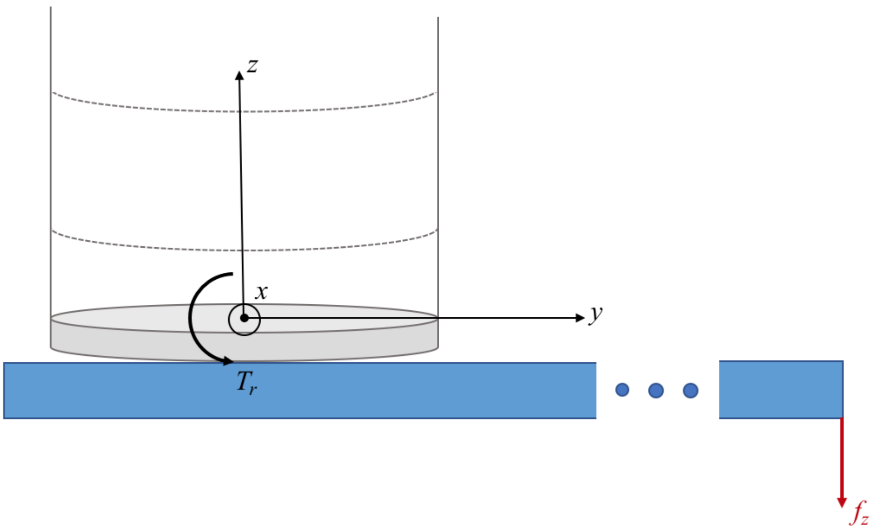


Figure 3.10: The manipulators end-effector attached at the left end of the steel frame that experiences a force f_z at its right end from the human. This causes a reactive torque T_r around the end-effector's x-axis.

When the collaborating human wants to move the frame in the negative z-direction it exerts a force f_z, shown in red in figure 3.10. Since the end-effector is rigid, this force f_z will cause a reactive torque T_r around the x-axis of the end-effector. However, this will also create forces in the positive z-direction exerting on the end-effector's left part of the surface in contact with the steel frame, viewing the end-effector and the steel frame as in figure 3.10. The human's descending movement of the steel frame can consequently be interpreted as both a torque on the end-effector as well as a force.

Two different experimental methods of control are defined below, the first one uses an angle-based approach in which the object is manipulated along the z-axis by tilting it (rotating around x-axis). By using this angle it can be determined if the collaborating human wishes to lift the object or not. The second method uses a constant lifting force exerted from the manipulator to help the collaborating human when lifting. This method uses the torques exerted upon the end-effector to decide whenever the manipulator should move the object along the z-axis or not. Both methods strive to use the same type of control for manipulation in the xy-plane that is specified below.

3.2.1 Angle-based controller

The angle-based controller is based on reading the angle, θ_x, that the operator creates around the x-axis in TCP-space by moving their end of the object in the positive or negative z-direction, see figure 3.11. The operator does this movement when he/she intends to raise or lower the steel frame in the z-direction during the lifting task.

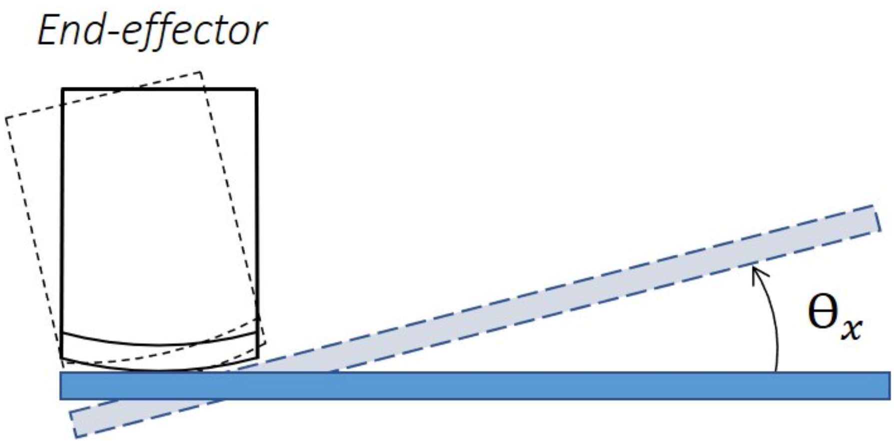


Figure 3.11: Angle deviation, θ_x, of the end-effector when the operator is tilting the object.

The matrices used to control the object by traditional admittance control can be defined as:

$$\bar{v} = \bar{C}\bar{f} \tag{3.1}$$

with

$$\bar{v} = \begin{bmatrix} v_x & v_y & v_z & \omega_x & \omega_y & \omega_z \end{bmatrix}^\intercal$$

where $v_x,\ v_y,\ v_z$ denotes the translational velocities and $\omega_x,\ \omega_y,\ \omega_z$ denotes the angular velocities around the x-, y- and z-axes. These velocities correspond to the degrees of freedom of the UR10, meaning that the $\bar{v}$ vector can be sent to the robot as a velocity based command. $\bar{C}$ and $\bar{f}$ are defined as:

$$\bar{C} = \begin{bmatrix} c_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_6 \end{bmatrix}, \bar{f} = \begin{bmatrix} f_x \\ f_y \\ f_z \\ t_x \\ t_y \\ t_z \end{bmatrix} \tag{3.2}$$

where $\bar{C}$ denotes the axes along which movement is allowed and if the movement is to be converted by some factor c_i for $i = 1, .., 6$. The factor c_i is an dampener and speed limiter that converts the measured force at the end-effector to a movement of the end-effector, in the same direction as the applied force while also limiting the the maximum speed to a user defined value and putting a damping on to ensure that the manipulator stops when no force is measured at the end-effector. $\bar{f}$ denotes the vector that describes the measured wrench at the end-effector of the manipulator, f denotes the force and t denotes the torque.

If force is applied along f_z and c_3 is nonzero, the manipulator will translate along the z-axis with a speed dependent on the values f_z and c_3. This is inefficient since the goal is to constrain the manipulator along the z-axis so that it does not drop the object it is holding. To counteract this the degree of freedom must be constrained while still maintaining the possibility of manipulating the object along all of the axes in $\bar{f}$. To achieve this, $\bar{v}$ is redefined as:

$$\bar{v} = v_{max}\bar{K}sgn(\bar{o}) + \bar{C}\bar{f} \tag{3.3}$$

$$\bar{K} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \bar{o} = \begin{bmatrix} x_{tcp} \\ y_{tcp} \\ z_{tcp} \\ \theta_x \\ \theta_y \\ \theta_z \end{bmatrix} \tag{3.4}$$

$$\bar{C} = \begin{bmatrix} c_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_6 \end{bmatrix}, \bar{f} = \begin{bmatrix} f_x \\ f_y \\ f_z \\ t_x \\ t_y \\ t_z \end{bmatrix} \tag{3.5}$$

where $\bar{o}$ is the vector containing the coordinates of the TCP defined in the TCP-space and sgn denotes the sign function. θ_x is shown in figure 3.11. v_{max} is a scalar value and corresponds to the maximum velocity that the TCP is allowed to reach along each independent axis, this value is set dependent on the implementation. $\sigma(\cdot) \in [0, 1]$ is set dependent on the size of the deviation in θ_x, see (3.6). Note that only the sign of θ_x value is of relevance, not the value itself. Meaning that a large deviation in θ_x will result in the same v_z as a smaller one.

By using this redefinition, the manipulator can be controlled according to equation (3.1) which will use "traditional" admittance control in all axes except z in TCP-space. Movement along the z-axis is instead achieved by the human by deviating θ_x to a nonzero value, see figure 3.12. A value of θ_x that is positive will make the end-effector's TCP move along the z-axis in a positive direction and a negative value will make it move in a negative direction. The movement will continue until the angle deviation equals zero. This deviation is created by inducing a torque, t_x that will create a velocity ω_x, which changes the value of θ_x, resulting in a nonzero v_z value. Note that if the human collaborator lifts their end of the object by a force f_{zH}, the end-effector will feel this as a torque $f_{zH}D_{HM}$ where D_{HM} denotes the distance between the human collaborator and the end-effector.

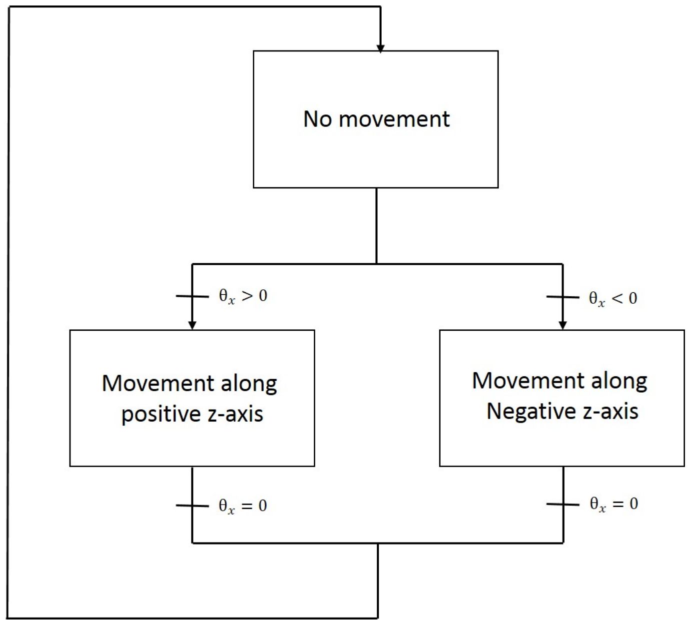


Figure 3.12: Sequential function chart describing the algorithm for movement along z-axis.

In order to make the operation feel more fluent to the human collaborator the following term is added:

$$\begin{bmatrix} -\theta_{x0} < \theta_x < \theta_{x0}, & \sigma = 0 \\ \neg(-\theta_{x0} < \theta_x < \theta_{x0}), & \sigma = 1 \end{bmatrix} \tag{3.6}$$

This creates an angular space in which the end-effector will not manipulate the object if within θ_{x0}, this is to avoid jerky motions and oscillating behaviour from the manipulator, see figure 3.13.

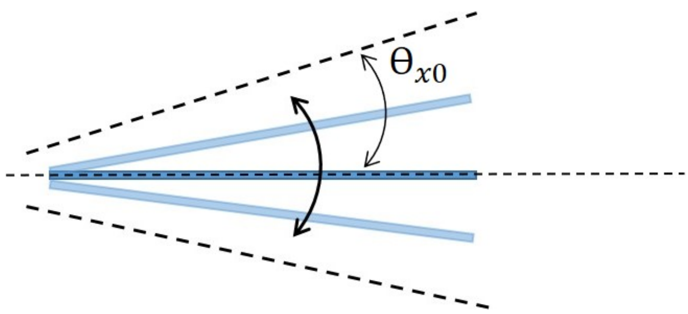


Figure 3.13: Angular space in where the manipulator will not react to change in θ_x.

c_1, c_2, c_4, c_5, c_6 and θ_{x0} are to be set so that the human working in collaboration with the manipulator perceives the movements as natural, this will vary dependent on the implementation.

An issue with using control based on the θ_x angle is that starting and ending the lift on flat surfaces can prove to be difficult to the human collaborator since it is difficult to create an angle deviation in θ_x when the object is lying down flat. This is because the pivot point of the manipulator is not at the end of the object. In this project this has been solved by adding an additional operation in the beginning and ending of a lift. When an input is given from the human collaborator that a lift is ready to start, the manipulator will grip the object and move a short distance linearly along the positive z-axis in the TCP-space. After this movement is concluded, the operator can manipulate the object freely.

Regarding the ending of the lift this has been solved by redefining $\bar{v}$ to include a constant velocity d_c along the negative z-axis. This results in:

$$\bar{v} = \begin{bmatrix} v_x & v_y & v_z - d_c & \omega_x & \omega_y & \omega_z \end{bmatrix}^\intercal \tag{3.7}$$

$$\begin{bmatrix} -\theta_{x0} < \theta_x < \theta_{x0}, & d_c \neq 0 \\ \neg(-\theta_{x0} < \theta_x < \theta_{x0}), & d_c = 0 \end{bmatrix} \tag{3.8}$$

This allows the operator to set their end of the object down onto the target surface and then wait for the manipulator to set its end of the object down, see figure 3.15.

When the manipulator senses a force along the positive z-axis (the object touches the target surface) it will set $d_c = 0$ and set the constants $\sigma = c_4 = c_5 = c_6 = 0$ in equation (3.4) and (3.5), to only allow translational movements in the xy-plane in TCP-space for small adjustments. When these adjustments have been made, the operator can release the tool from the frame by pushing the end-effector down with a force $f_{z,start}$, see figure 3.14. This will create a reacting force measured at the end-effector in the positive z-direction, which will enable the push from the operator to be measured by the force sensor and activate the releasing operation depicted in figure 3.15. The same method is used for starting the gripping operation where after the operator has moved the tool to the desired grip position they will push the end-effector down with a force which will start the gripping operation.

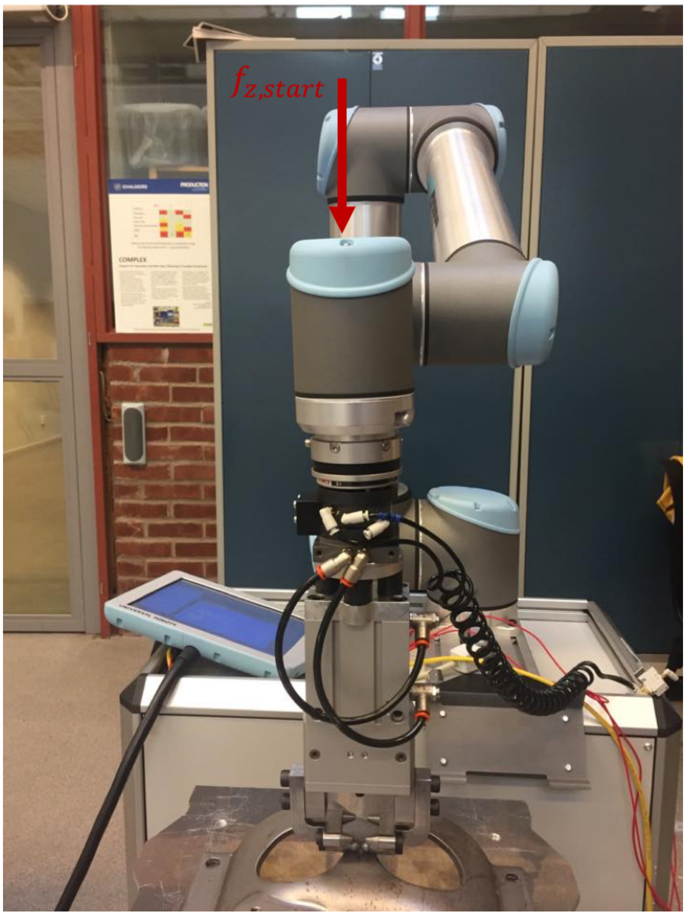


Figure 3.14: The force $f_{z,start}$ that the operator exerts on the manipulator in order to activate the releasing operator.

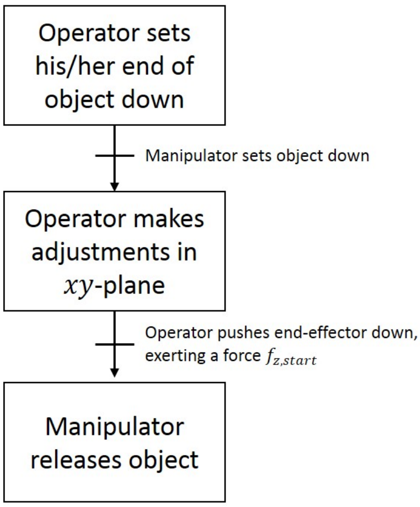


Figure 3.15: Depiction of the ending of a lifting operation.

3.2.2 Force/torque-based controller

An alternative approach is to have a controller that is more dependent on the forces and torques exerted at the end-effector for movement along the z-axis in TCP-space. Here, a constant driving force f_c that the manipulator exerts on its end of the object is defined, see figure 3.16. Note that f_z is the force that the end-effector is measuring and reflects the force that the operator affects the object with.

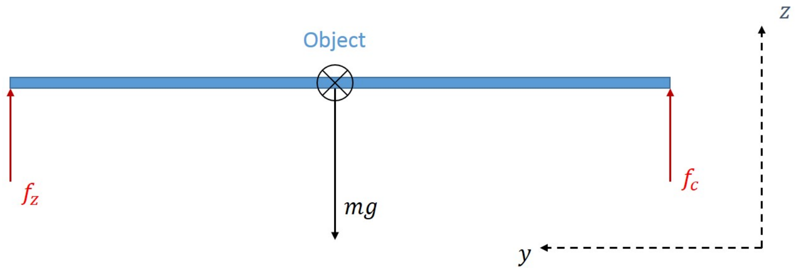


Figure 3.16: Depiction of the forces affecting the object in the z-direction.

Using figure 3.16 it can be determined using Newton's second law that the force f_z that the human collaborator needs to exert upon the object for it to have no velocity along the z-axis is:

$$f_z = mg - f_c$$

Here m denotes the mass of the object and g is the gravitational constant. For creating accelerations, the following equations are defined using Newton's second law:

$$\frac{1}{m}(f_z + f_c) - g = a \tag{3.9}$$

where a is the acceleration. The equation describes movement along the z-axis of the end-effector. This means that the direction of the movement along the axis will depend on f_z being greater or lesser than $mg - f_c$, assuming that $f_c \leq mg$.

Using (3.9), a discrete model can be defined using the Forward-Euler [17] method of approximating the movement along the z-axis based on the forces f_c and f_z:

$$v_{z_{n+1}} = v_{z_n} + \int_{t_n}^{t_{n+1}} \left(\frac{f_z + f_c}{m} - g \right) dt, v_z(0) = 0$$

where v_{z_n} is the velocity along z in TCP-space, t_n is a time step that is discretizied when implemented on the real controller.

The matrices from (3.2) are used, but redefined slightly. The new matrices are:

$$\bar{v} = \bar{C}\bar{f} + \bar{K} \tag{3.10}$$

with

$$\bar{v} = \begin{bmatrix} v_x & v_y & v_z & \omega_x & \omega_y & \omega_z \end{bmatrix}^\intercal$$

$$\bar{C} = \begin{bmatrix} c_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_6 \end{bmatrix}, \bar{f} = \begin{bmatrix} f_x \\ f_y \\ f_z \\ t_x \\ t_y \\ t_z \end{bmatrix}, \bar{K} = \begin{bmatrix} 0 \\ 0 \\ v_{z_{n+1}} \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.11}$$

f_c is set by weighing the object at the beginning of the operation and setting f_c to an experimental initial value that is 75% of the measured weight of the object. An important note is that $c_4 = c_5 = 0$ to constrain the manipulator from rotating around the x- and y-axes so that the manipulator can read the forces exerted upon it more clearly. When a torque t_x is felt by the force/torque sensors the value of f_c will be updated using the Forward-Euler method:

$$\begin{cases} f_{cn+1} := f_{cn} + c_t, & t_x > 0 \wedge f_c < 160\text{N} \\ f_{cn+1} := f_{cn} - c_t, & t_x < 0 \wedge f_c > 70\text{N} \end{cases}$$

where $c_t = 1$N is the increase or decrease in the force based on t_x and the updating frequency is 10Hz. The intended function is to sense if the f_z force is negative or positive (the operator is pushing the object along the z-axis), since the f_z force will create a torque t_x perceived by the force/torque-sensor at the end-effector. Adjusting the f_c force makes the apparent mass of the object seem lighter to the operator. f_c is bounded as 70N$< f_c <$ 160N so that the safety of the operator is not compromised.

3.2.3 Switching between translating and rotating movement

In this project, an attempt has been made to create a solution that uses an algorithm which will change between modes through analysing input, rotation around the z-axis and translation in the xy-plane, both in TCP-space. This is achieved by using a similar method to Karayiannidis et al. [1], where the decision of which mode is to be used is decided by the absolute value of the force exerted at the end-effector. At lower forces, the object will rotate and if a larger force is applied the object will instead translate. This enables the collaborative human to decide what mode the manipulator will use only by manipulating the jointly held object. This method proves to be quite smooth and simple to understand for an inexperienced operator.

Since the the manipulator only uses force sensors located at the end-effector to sense the collaborating human's movements, it will have difficulties in distinguishing between exerted forces and torques from the operator assuming that the operator only has one contact point with the object [2]. Therefore, to have the possibility to both rotate and translate the object in a horizontal direction, the following method is used. A constant force constraint f_{sw} is defined. This constant is used as a way to distinguish between when translating and rotating movement should be permitted by the manipulator, see figure 3.17. f_{xH} denotes the x-direction of the human applied force at his/her grasping position.

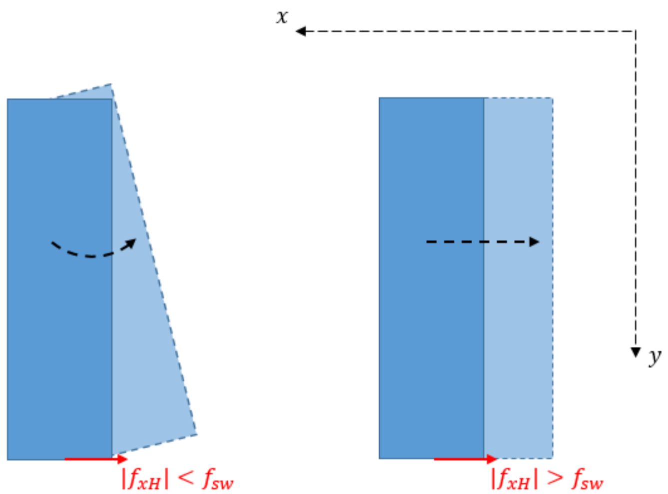


Figure 3.17: Image of the force f_{xH} exterted by the human in the x-direction causing the object to either rotate or translate.

The definition becomes:

$$\begin{bmatrix} |f_{xH}| < f_{sw}, & c_1 := 0 \\ |f_{xH}| > f_{sw}, & c_6 := 0 \end{bmatrix} \tag{3.12}$$

This ensures that the human collaborator has a way of communicating whether the goal is to translate or rotate the object in a simple way. If the operator wishes to translate the object, he/she only needs to apply a firmer force upon their end of the object being manipulated and vice versa.

3.2.4 Implementation of admittance control

In the implementation of the control theory described above there were many factors to consider. Different methods of implementing the control were tested and by comparing the different methods a solution was produced. One issue that was found when implementing the admittance control was that the manipulator needs to respond quickly to the input from the collaborating human in order for the motion to feel natural. This means that a fast system without delays or loss of information is necessary. When using an external device that sends commands continuously to the manipulator through Robot Operating System, the sampling time using ROS through a virtual image of Linux was between 5Hz and 10Hz. Andersen [11] notes that a sampling rate of at least 15Hz is needed to avoid jerky motion, therefore this method was deemed inadequate. There are possibilities to increase the sampling rate by using faster hardware or rewriting the driver *(UR modern driver* [11]) that is already written and commonly used when controlling the UR10 through ROS. Another option is to run ROS on a native installation of Linux instead of through a virtual machine. This has been observed to speed up communication with the manipulator up to 125Hz. The main benefit of this external device is the large flexibility it provides in reading data and implementation in larger systems that rely on communication between each other.

Another approach is to implement the algorithm using only the operating system of the manipulator. This is achieved by writing programs using the UR10 teach pendant. The benefits of this is that the refresh rate is 125Hz [8], which makes the lifting operation feel more fluent to the collaborative human. This solution is limiting because it is implemented using offline programming only, meaning that if the end user seeks to implement across multiple manipulators or in some other way wishes to have more control over the system, it may prove challenging. Partly because the solution becomes limited to the UR10 specifically, but also that it limits communication with external devices.

The solution that was chosen was to combine the two different methods described above to try and utilise the benefits of each of them. The architecture depicted in 3.18 describes the final system. The operations in the sequential function chart in figure 3.1 are all executed locally on the UR10's internal operating system and the external device is used as a master that reads sensor data to determine when the switches between states in 3.1 are to occur. This method provides the refresh rate required for the lifting algorithm to run smoothly, while also keeping the flexibility of having an external master that is capable of communication between multiple systems.

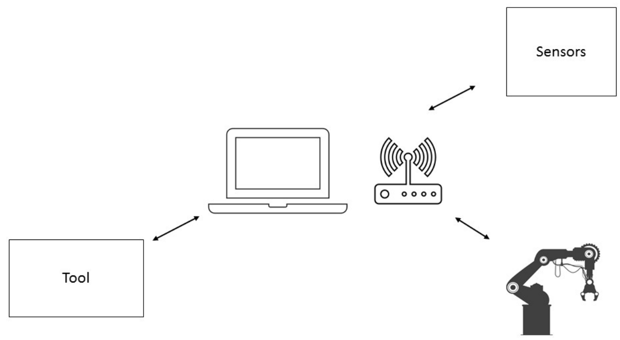


Figure 3.18: System architecture.

In this project, the algorithm is implemented by using Python and ROS on the external master to communicate with the robot via *UR modern driver* written by Thomas Andersen [11]. The communication is done via TCP socket programming and supports the option of wireless implementation. The master also has external force/torque sensors connected to better assess which state the manipulator should be in. When a change of state happens the master sends a request to load a new program on the manipulators operative system that will start and then run locally to make use of the faster response time discussed above. If the program fails to load, the robot will go into a fail-safe mode and will need a confirmation from the user that the operation is ready to restart.

Regarding the lifting algorithm, software that the UR10 uses already has certain commands implemented that are relevant to this application. One of these commands is *force-mode* [8] which lets the user define which axes the manipulator should be compliant along. It also lets the user define forces that the manipulator will exert along certain axes. Using this command allows implementation of the control theories stated above without using an external computer for control. In this case, compliance along $\begin{bmatrix} x & y & r_x & r_y & r_z \end{bmatrix}$ while regulating z based on the angle θ_x achieves the results wanted for lifting and manipulating the object when using the angle-based controller. When implementing the force/torque-based controller, force-mode is used to define the driving force f_c along the z-axis of the TCP while constraining movement along r_x and r_y. x, y and r_z are left fully compliant.

4 Results and discussion

In this section the controllers will be compared and analysed. The tests that were made on the controllers will be presented along with their results.

4.1 Testing and verification

In order to test the angle-based algorithm, the force/torque-based algorithm and the algorithm for translation and rotation, some experiments were conducted. During these experiments, we gathered data about how the manipulator reacts to different inputs from the human collaborator. The manipulation along the z-axis of the TCP, by deviating the angle θ_x, was tested. Another test was done to examine the functionality of the solution for the translation versus rotation problem. A test with an inexperienced operator was performed to further test the intuitiveness of the force/torque algorithm. The angle-based algorithm was chosen not to be tested with an inexperienced operator for safety reasons.

4.1.1 Angle-based controller

This test was performed by manipulating the object linearly in the TCP-space along the z-axis. Two tests were performed where the permitted maximum acceleration a_{max} was altered in order to see how it would change the manipulator's behaviour. The test was done by deviating the angle θ_x and observing the object's translation in the z-direction. Figure 4.1 shows the result of the test with $a_{max} = 0.2\text{m/s}^2$ and figure 4.2 shows the result of the test with $a_{max} = 0.8\text{m/s}^2$. Note that the angular space θ_{x0} is marked in the upper graphs of figure 4.1 and 4.2, and define the space where the manipulator will not move along z. The vertical lines represent when the operator deviates θ_x enough that translation will commence. The time $t_{v_{max}}$ is the time measured in seconds that it takes for the manipulator to accelerate in order to reach its maximum velocity $v_{z_{max}} = 0.2\text{m/s}$ along z.

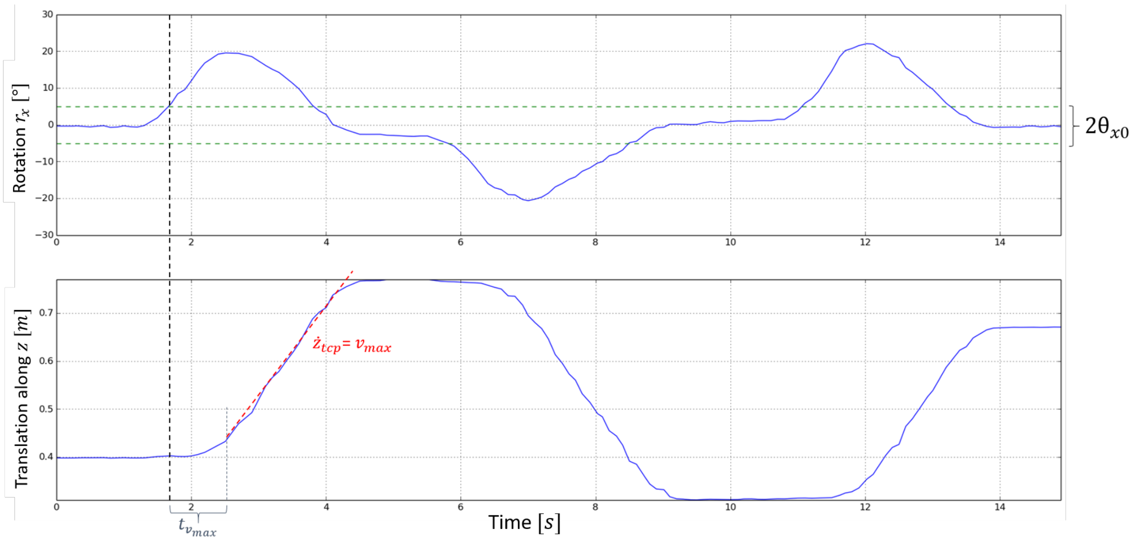


Figure 4.1: Angle-based algorithm with $a_{max} = 0.2\text{m/s}^2$, $t_{v_{max}}$ is measured to 0.85s.

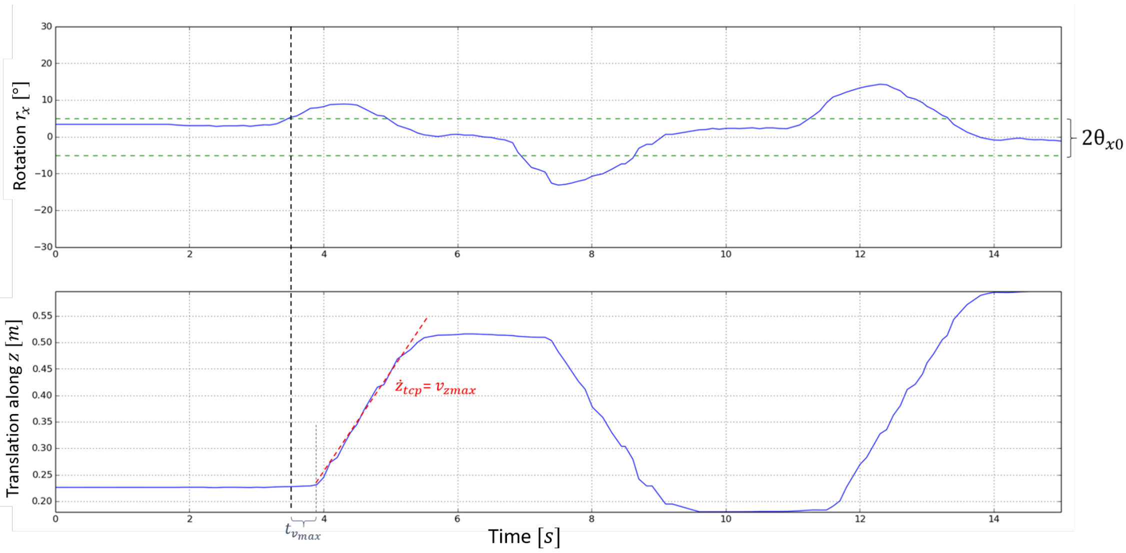


Figure 4.2: Angle-based algorithm with $a_{max} = 0.8\text{m/s}^2$, $t_{v_{max}}$ is measured to 0.375s.

In the tests it was noted that the manipulator reacted faster and the operator did not create a significant angle deviation in θ_x when a higher a_{max} was used. This because the manipulator lifted its end of the object to the same height as the operator faster due to the higher acceleration. The manipulation being faster may increase the fluidity of the operation to the collaborating human, however, from a safety perspective, it might be desirable to limit the acceleration of the TCP as it will take a larger toll on the manipulator at high accelerations while simultaneously feeling more intimidating to the operator. It is also of importance to set $v_{z_{max}}$ to a value smaller than the speed that the operator lifts their end of the object with, else the end-effector could catch up to the operator before the he/she has reached the desired position along z, which would be perceived as the manipulator moving in a jerky fashion. Therefore the $v_{z_{max}}$ was limited to 0.2m/s.

4.1.2 Force/torque-based controller

The test was performed similarly to the one performed on the angle-based controller. An object was manipulated along the z-axis of the TCP using the force/torque-based controller. The manipulator reacts mainly to deviations in the torque around the x-axis, t_x, measured at the end-effector and uses this to increase or decrease f_c. The results are depicted in figure 4.3

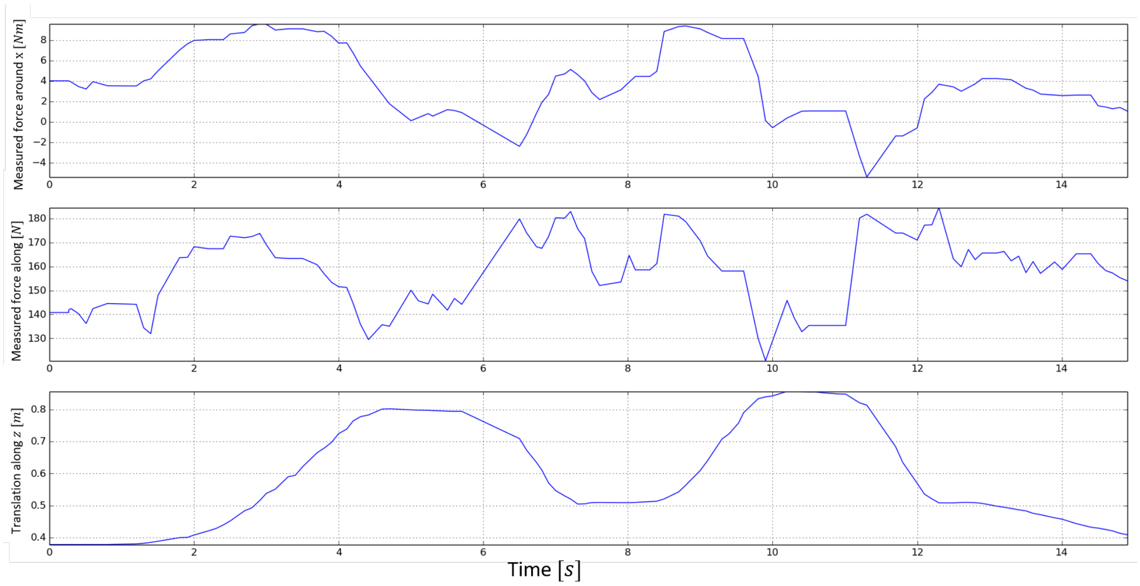


Figure 4.3: Force/torque-based controller.

By comparing the first and second graph of figure 4.3, it can be seen that the force increases when the torque increases and vice versa. Comparing the first and third graph shows that the torque decreases as the height gets closer to the end position where the change in z is close to 0. From comparing the second and third graph the force needed to achieve a resting position seem to be around 140-150N. It is worth noting that the second graph seem to have some interference in the measurement of force, which is responsible for the large spikes in change of force.

This method is not completely accurate when it comes to reading the intention of the operator since an increase or decrease of the t_x value does not necessarily mean that the operator wishes to translate the object.

4.1.3 Translation and rotation

To test the algorithm's capability in dealing with translating and rotating movement, the movement in the z-direction was restricted and only the algorithm that dealt with translating and rotating movement was tested. A test was done that consisted of three events, e_1, e_2 and e_3. The events e_1 and e_2 demonstrates the operator translating the object along the x-axis of the TCP. e_3 demonstrates a rotating movement around the TCP z-axis. The results are depicted in figure 4.4.

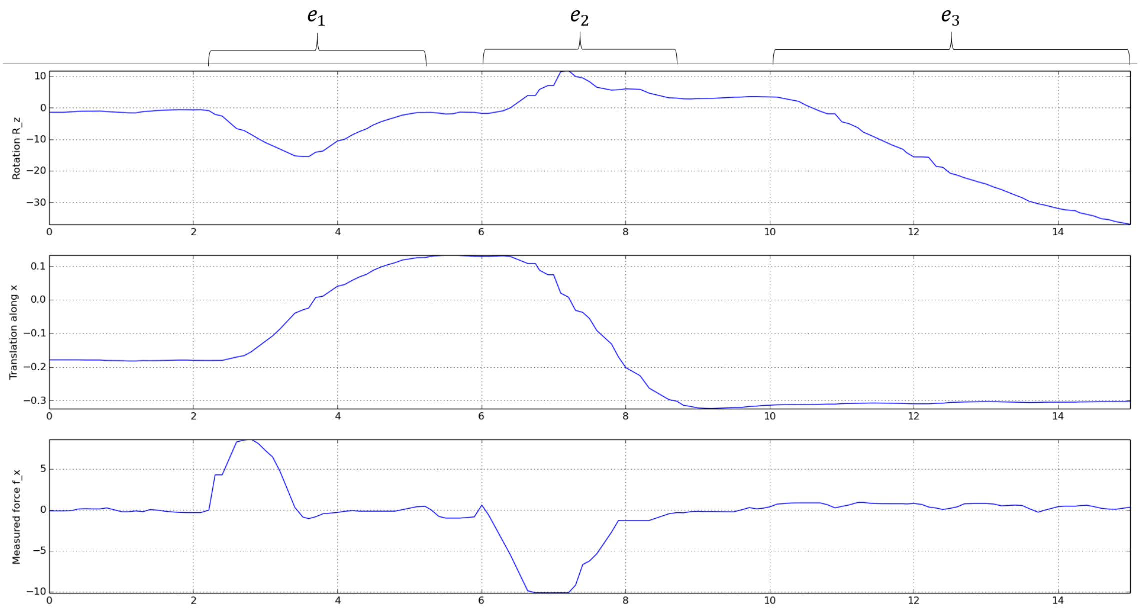


Figure 4.4: Graph showing translating and rotating movement, divided in three events e_1, e_2 and e_3, based on the force f_x measured at the end-effector.

From the test it can be seen that if a larger force f_x is felt by the end-effector, it will set $c_6 = 0$ and thus restrict rotating movement around the z-axis of the TCP. If the operator wishes to rotate the object, he/she can do so by not applying any large force f_x, which can be seen in event e_3.

4.1.4 Force/torque-based controller: testing with an inexperienced operator

In order to test the force/torque-based algorithm's function and ease-of-use, an inexperienced operator was tasked with lifting a frame from one surface and to place it onto a mock-up of an engine. The operator was given basic instructions how the algorithm operates and how to translate and rotate the frame. He was also told how to begin and end the operation.

The operator experienced that the manipulator felt the operator's movements of the frame and that it followed them. The payload of the steel frame was felt like it was shared equally between the operator and the manipulator. When the inexperienced operator compared performing the assembly task collaboratively with the manipulator and without it, it was easier and felt more ergonomic to lift together with the manipulator. One problem that the operator experienced was that the manipulator had some difficulties sensing when the operator wanted to move the frame away from himself and towards the manipulator, in the negative y-direction. Since the pneumatic tool is rather long in the z-direction, every force the operator exerts on the metal frame will be interpreted as a torque. This, combined with the fact that the tool will bend when high forces are applied, makes it very difficult for the manipulator to determine if the operator wants to lift the frame or move it along the y-axis.

Further results of the test with the inexperienced operator were that he experienced the assembly task to be easier performed with both hands than with one. He then placed one hand at the end of the steel frame and the other close to the middle of the steel frame. This made it more simple to have control of the steel frame and manipulate it.

4.2 Comparing the two different controllers

This project has developed two different controllers for manipulating the object in collaboration with a human. The two algorithms used by the controllers are different in nature and they have different advantages based on the setting and implementation.

4.2.1 Angle-based controller

This algorithm was developed under the premise that the tool provided during the initial phase of the project (magnetic tool) did not handle torques very well and therefore an effort was made to minimise the torques that would be exerted upon the end-effector of the manipulator. Allowing compliance along r_x, r_y and r_z in TCP-space means that the human operator could not exert any significant torques, as it would rotate instead of creating reactive torque, which was desirable.

One advantage of this controller is that it is relatively easy for the human collaborator to move the object in the xy-plane of the TCP since if the frame is kept within the angular space θ_{x0} depicted in figure 3.11, the object is constrained to movement in the xy-plane meaning that unintentional movements along the z-axis of the TCP are not likely. If the operator wishes to do precise positioning of the object in the xy-plane, this method might be superior to the force/torque-based controller. It is also noted that this controller proves more efficient if the operator wishes to manipulate the object by only gripping the end of it, since it is easier to create the t_x torque around the TCP because of the increased length of the lever between the end-effector and the force the collaborating human exerts.

When the pneumatic tool was introduced later in the project it allowed for greater torques to be exerted upon the end-effector because of the more reliable grip upon the object. It also had more mass and offset the TCP because of its length compared to the magnetic tool. This proved problematic to the manipulator and it was observed that protective stops occurred frequently because of high perceived forces exerted upon the joints of the manipulator.

The rubber spacings present on the pneumatic tool seemed to introduce oscillating behaviour from the end-effector, since they would compress when the operator lifts the object and then decompress which would make the end-effector rotate around the x-axis in TCP-space. This causes another compression of the spacings, thus leading to an oscillating behaviour. This can be resolved by simply removing the rubber spacings or by restricting the permitted velocity around the x-axis of the manipulator, which would act as a dampening element to reduce the settling-time of the oscillations. However, this could also compromise the fluidity of the movement perceived by the human collaborator.

4.2.2 Force/torque-based controller

This algorithm was designed to work when using the pneumatic tool and therefore there was no real limit to how much force or torque the tool would need to withstand. Therefore, this method uses more direct measurements from the force/torque-sensors at the end-effector to calculate movement. The issue of reading the intention of the operator became more apparent when using this method of control since there is no real framework in place for differentiating between different types of movement. An example being movement in the xy-plane of the TCP. In the case of the angle-based controller, the manipulator was restricted to movements in the xy-plane as long as the object was not being tilted noticeably. This meant that movement only in this plane could be achieved by keeping the object level between the operator and the manipulator, while manipulating the object freely in the xy-plane. In the case of the force/torque-based controller, the intention to manipulate the object in the xy-plane was harder to differentiate from linear motion along the z-axis of the TCP due to the measured forces at the end-effector. If the operator pulls the object towards them, this can also be perceived as an intention to move the object along the z-axis because of the lever l_t that is present between the force/torque sensor at the end-effector and the point where the tool grips the object, see figure 4.5.

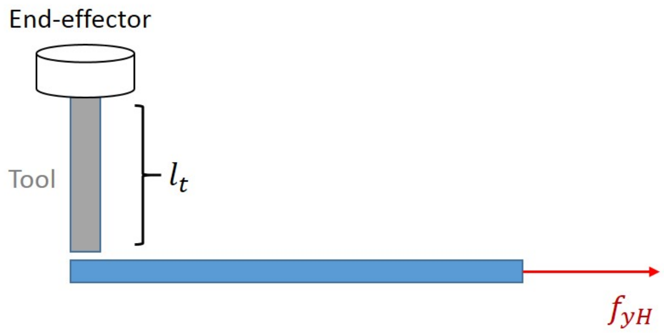


Figure 4.5: Depiction of lever, l_t, introduced by the tool.

Because of this, it was noted that if the operator would grip the object with two hands it was easier to communicate the operators intention to the manipulator. With one hand gripping the end of the object and one closer to the middle of the object, the operator had more control over the direction and orientation of the object. When this was done, the operator could quite easily manipulate the object and the motion felt reasonably natural. This separates the two different controllers in the sense that when the angle-based controller is used, it is easier if the operator is only gripping the end of the object with both hands due to the increased lever introduced by the length of the object. Theoretically, the angle-based controller shares the load evenly between the operator and the manipulator while the force/torque-based algorithm is more suitable for applications where the goal is to reduce the force that the operator needs to exert upon the object. This can be done by increasing the driving force f_c that the manipulator exerts upon the object. This implies that these two controllers can be useful in different situations.

4.3 Safety

The manipulator is intended for handling tools and is classed for collaborative operations [8]. This means that it has special safety-related features in order to work close together with a human without fences or other types of barriers. One of these features is the protective stop, also called a *Stop Category 0*, that is used when a limit is exceeded or a fault has occurred in the control system that is related to safety. When this happens, each joint of the manipulator stops and the robot loses power. On the teach pendant of the manipulator, there is an emergency stop button that can be pushed at all times in order to immediately stop all robot motion. This also causes the joints to stop and the power to the robot is cut.

Because of the manipulator being able to handle forces up to 250N and having the maximum recommended payload of 10kg, there is a safety risk for the collaborating human when the exerted forces on the manipulator are getting close to those maximum values [8].

When it comes to the humans safety in collaboration with the robot, the safety of the tools can be discussed. The magnetic tool can hold on to the steel frame until it senses a torque that is too big and it drops the frame. This sensitivity to torque together with being sensitive to the surface on which the tool is being placed, makes it rather unsafe for the collaborating human since the heavy metal frame might be dropped on the human followed by the manipulator increasing or decreasing its position along the z-axis. The pneumatic tool will only release its grip of the steel frame through an output from the teach pendant and will therefore not drop the frame unexpectedly, as the magnetic tool might. Another thing worth noting is the risk of impingement when the pneumatic tool closes its contraption in order to grip the frame. It is therefore of importance that the operator does not keep his/her hands close to the contraption part of the tool during this sub-task.

The force/torque algorithm, which the pneumatic tool uses, is less safe than the angle-based. When running the force/torque based algorithm, the manipulator will sometimes suddenly drag the metal frame in the x- or y-direction with a large force when there is only supposed to be able to create forces along the z-axis. This might be due to the transform and position data from the manipulator not being perfect so its coordinate system will be slightly angled and it will then push in x- or y-direction while believing it is only lifting it in the z-direction. The pneumatic tool is the safer tool, but is accompanied by an unsafe algorithm. The best result would come if the angle-based algorithm was used with a pneumatic gripper without the flexible rubber parts. This would allow for the use of the algorithm without the oscillating behaviour.

Further improvements of the safety are possible when the environment, where the manipulator is going to operate in, is known. Knowing this would allow restrictions in the algorithm for where the manipulator is allowed to move its end-effector. This could ensure that it will never be in face or knee height, which are positions where the operator has less strength to control the manipulator.

4.4 Social and ethical aspects

Within this project, there are a number of social and ethical aspects that need to be considered. When the level of automation within factories is increased, it might affect the employees in both a positive and negative way. The increased level of automation could reduce the amount of jobs available within production because of manpower being transferred from humans to robots. On the other hand, it might create new job opportunities that comes with robots, for example robot development and maintenance. However, this project deals with a relatively small assembly task compared with an entire production and will therefore probably not have a significant effect on job opportunities in factories.

The task to assemble collaboratively with the human that is given to the manipulator, seeks to improve the ergonomic aspect of the assembly task. This collaborative assembly will improve this aspect due to the reduced payload of the steel frame that the human will lift.

Because of the human and the robot working in such close proximity during the assembly task, the safety of the human is of great importance. Although the robot that is used within this project is classed for working in collaboration with humans without safety fences, the risks of damage to the human must be minimised.

5 Conclusion

This project has tried to solve the problem of manipulating a jointly-held object in collaboration with a manipulator. This has lead to the evaluation of two different tools for gripping the object and two different controllers for manipulating in collaboration with an operator. Tests showed that the pneumatic tool was much safer with respect to the firmness of the grip and minimising the risk of dropping the object. This tool introduced problems however, since it had some softer parts attached that could cause oscillating behaviour when introduced to forces and torques. It was concluded that these parts should be removed to increase the lifting experience from the operator's viewpoint.

The angle-based controller is more reliant when it comes to manipulation in the horizontal xy-plane and handles translations along z-direction more fluent when the manipulator is not exposed to the weight of the frame. When the frame is attached, the mass paired with the pneumatic tool's mass is proven to be difficult for the manipulator to handle without oscillating movements around the x-axis of the TCP. As mentioned, the removal of the softer parts from the tool may help remedy this problem.

The force/torque-based controller proves to be more successful in handling the combined mass of the frame and the pneumatic tool. This controller, however, has difficulty in interpreting the movements of the operator by only reading forces and torques at the end-effector. It was noted that the manipulation is perceived as easier to the operator if the object is held with one hand at the end and one closer to the middle of the object, since this allows the operator to create torques easier. The group concludes that the implementation of this controller is more suited as a way to make an assembly task more ergonomic to the operator but is not suited for precise object positioning.

5.1 Further development

In the future the solution can be further developed in order to make the mounting task more flexible and applicable to the production. In order to increase the ergonomics within the task the manipulator can utilise its full lifting capacity and decrease the human's load. Since the maximum lifting capacity of the manipulator is 10kg and the steel frame weights 13kg, the manipulator can carry more than half of the weight from the steel frame. However, it needs to be taken in account that the tool mounted on the manipulator has its own weight and must be included in the total payload of the manipulator.

There has not been any development of a specific safety system for the human while executing the task. Although some consideration has been taken regarding the safety of the human during the mounting task, a full safety system could be developed in the future. This system could involve both the manipulator and the tool for the human to be able to work in close proximity with the robot without compromising on safety.

In order to make the solution more flexible there can be added a possibility of changing tools on the manipulator automatically. The produced solution requires the operator to manually attach the tool to the manipulator, which could be automated. This could make the operation more efficient if the solution was to be implemented into production. Since the pneumatic tool is controlled (attached, detached, opened, closed) by compressed air, a future improvement is to control it through an external control unit. A complement to this development is to implement a camera to the system. The camera could locate the desired tool within a specific area. This would make it possible for the manipulator to mount the tool automatically after the tool is detected. Camera could locate other objects as well, for example the steel frame.

Further, the pneumatic tool can be modified by removing the rubber spacers, alternatively, the tool can be redesigned so that it is shorter and more lightweight. This would enable the manipulator to handle more of the weight of the frame, it would also mean that forces and torques could be read more clearly since the tool would not introduce any significant lever between the end-effector and the TCP because of its length.

References

[1] Y. Karayiannidis, C. Smith, and D. Kragic, "Mapping human intentions to robot motions via physical interaction through a jointly-held object", in *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, IEEE, 2014, pp. 391–397, ISBN: 978-1-4799-6765-0. DOI: 10.1109/ROMAN.2014.6926284. [Online]. Available: http://ieeexplore.ieee.org/document/6926284/.

[2] T. Wojtara, M. Uchihara, H. Murayama, S. Shimoda, S. Sakai, H. Fujimoto, and H. Kimura, "Human–robot collaboration in precise positioning of a three-dimensional object", *Automatica*, vol. 45, no. 2, pp. 333–342, 2009. DOI: 10.1016/j.automatica.2008.08.021. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0005109808004615.

[3] A. Mörtl, M. Lawitzky, A. Kucukyilmaz, M. Sezgin, C. Basdogan, and S. Hirche, "The role of roles: Physical cooperation between humans and robots", *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1656–1674, 2012. DOI: 10.1177/0278364912455366. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364912455366.

[4] S. Yigit, C. Burghart, and H. Worn, "Co-operative carrying using pump-like constraints", in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 4, IEEE, 2004, pp. 3877–3882, ISBN: 0-7803-8463-6. DOI: 10.1109/IROS.2004.1390019. [Online]. Available: http://ieeexplore.ieee.org/document/1390019/.

[5] J. Dumora, F. Geffard, C. Bidard, T. Brouillet, and P. Fraisse, "Experimental study on haptic communication of a human in a shared human-robot collaborative task", in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2012, pp. 5137–5144, ISBN: 978-1-4673-1736-8. DOI: 10.1109/IROS.2012.6385721. [Online]. Available: http://ieeexplore.ieee.org/document/6385721/.

[6] T. Takubo, H. Arai, Y. Hayashibara, and K. Tanie, "Human-Robot Cooperative Manipulation Using a Virtual Nonholonomic Constraint", *The International Journal of Robotics Research*, vol. 21, no. 5-6, pp. 541–553, 2002. DOI: 10.1177/027836402321261904. [Online]. Available: http://journals.sagepub.com/doi/pdf/10.1177/027836402321261904.

[7] Universal Robots staff, *Polyscope manual*, Universal Robots A/S, 2017. [Online]. Available: https://s3-eu-west-1.amazonaws.com/ur-support-site/27541/Software_Manual_en_Global-3.4.4.pdf.

[8] U. R. staff, *User Manual UR10/CB3*, Universal Robots A/S, 2017. [Online]. Available: https://s3-eu-west-1.amazonaws.com/ur-support-site/27484/UR10_User_Manual_en_E67ON_Global-3.4.5.pdf.

[9] T. Dirk. (2014). Wiki.ros.org, [Online]. Available: http://wiki.ros.org/ROS/Introduction (visited on 04/17/2018).

[10] T. T. Andersen. (2015). Github ur_modern_driver, [Online]. Available: https://github.com/ThomasTimm/ur_modern_driver (visited on 05/08/2018).

[11] T. T. Andersen, "Optimizing the Universal Robots ROS driver.", Technical University of Denmark, Department of Electrical Engineering, Lyngby, Denmark, Tech. Rep., 2015. [Online]. Available: `http://orbit.dtu.dk/en/publications/optimizing-the-universal-robots-ros-driver(20dde139-7e87-4552-8658-dbf2cdaab24b).html` (visited on 04/17/2018).

[12] D. Goepp. (2015). Optoforce ROS driver, [Online]. Available: `https://github.com/shadow-robot/optoforce` (visited on 05/08/2018).

[13] A. R. Hambley, *Electrical engineering : principles and applications*, 6th ed. Upper Saddle River, NJ, United States of America: Pearson Education M.U.A, 2014, p. 726, ISBN: 9780273793250.

[14] R. Ikeura and H. Inooka, "Variable impedance control of a robot for cooperation with a human", in *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, vol. 3, IEEE, pp. 3097–3102, ISBN: 0-7803-1965-6. DOI: `10.1109/ROBOT.1995.525725`. [Online]. Available: `http://ieeexplore.ieee.org/document/525725/`.

[15] J. Buchli, F. Stulp, E. Theodorou, and S. Schaal, "Learning variable impedance control", *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 820–833, 2011. DOI: `10.1177/0278364911402527`. [Online]. Available: `http://journals.sagepub.com/doi/10.1177/0278364911402527`.

[16] K. Kosuge, H. Kakuya, and Y. Hirata, "Control algorithm of dual arms mobile robot for cooperative works with human", in *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat.No.01CH37236)*, vol. 5, IEEE, 2001, pp. 3223–3228, ISBN: 0-7803-7087-2. DOI: `10.1109/ICSMC.2001.972015`. [Online]. Available: `http://ieeexplore.ieee.org/document/972015/`.

[17] D. F. Griffiths and D. J. Higham, *Numerical Methods for Ordinary Differential Equations*. London: Springer, 2010, pp. 19–31, ISBN: 978-0-85729-147-9. DOI: `https://doi.org/10.1007/978-0-85729-148-6`. [Online]. Available: `www.springer.com/series/3423`.