Institute of Architecture of Application Systems

University of Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Bachelorarbeit Nr. 286

Fluent Teleoperating of sequential mobile manipulation with a PR2

Melvin Klein

Course of Study:	Softwaretechnik
Examiner:	Prof. Dr. Marc Toussaint
Supervisor:	M. Sc. Andrea Baisero
Commenced:	November 24, 2015
Completed:	May 24, 2016
CR-Classification:	I.7.9, H.5.2

Kurzfassung

Diese Arbeit befasst sich mit der Verbesserung einer bereits existierenden Teleoperation Anwendung, die dem Nutzer keinerlei Rückmeldungen gibt. Sie beinhaltet drei verschiedene verbesserte Versionen und eine Diskussion über deren Vor- und Nachteile. Eine Bewertung der finalen Version und die Probleme, die noch behoben werden müssen, sind auch Teil dieser Arbeit.

Abstract

This work focuses on the improvment of an existing teleoperation that gives the operator no feedback. It features three different improved versions and a discussion about their pros and cons. An evaluation of the final teleoperation version and the problems that need to be addressed are also part of this work.

Contents

1 Introduction 9
1.1 Motivation 9

2 Related work 13

3 Discussion 15
3.1 Starting situation 15
3.2 Fixed movement restrictions 16
3.3 Force constraints 18
3.4 Flexible movement restrictions 18

4 Evaluation 21

5 Problems 25

6 Conclusion and future work 27

Bibliography 29

List of Figures

1.1 The lockbox I used to test the precision of the teleoperation. 9
1.2 The PR2 which was used to perform the teleoperation. 10

2.1 The SAM exoskeleton used by the ESA. 13

3.1 The left glove with the Polhemus G4 position tracking system attached to it. 15
3.2 The gamepad used to initialize and exit the teleoperation. 16
3.3 The axis of rotation on the robots gripper on the left and the axis of rotation on the human hand using the rotation of the elbow joint on the right. 19

5.1 The robot in a joint state which prohibits further movement. 25

1 Introduction

1.1 Motivation

When working with robots you often want them to interact with objects around them. In order to achieve this, you can either have the robot interact with objects by itself, or you have to give the movement commands manually. In a static environment you could hard-code the movement of the robot in order to make it interact with objects. In a dynamic environment this wouldn't work. You would have to use a machine learning

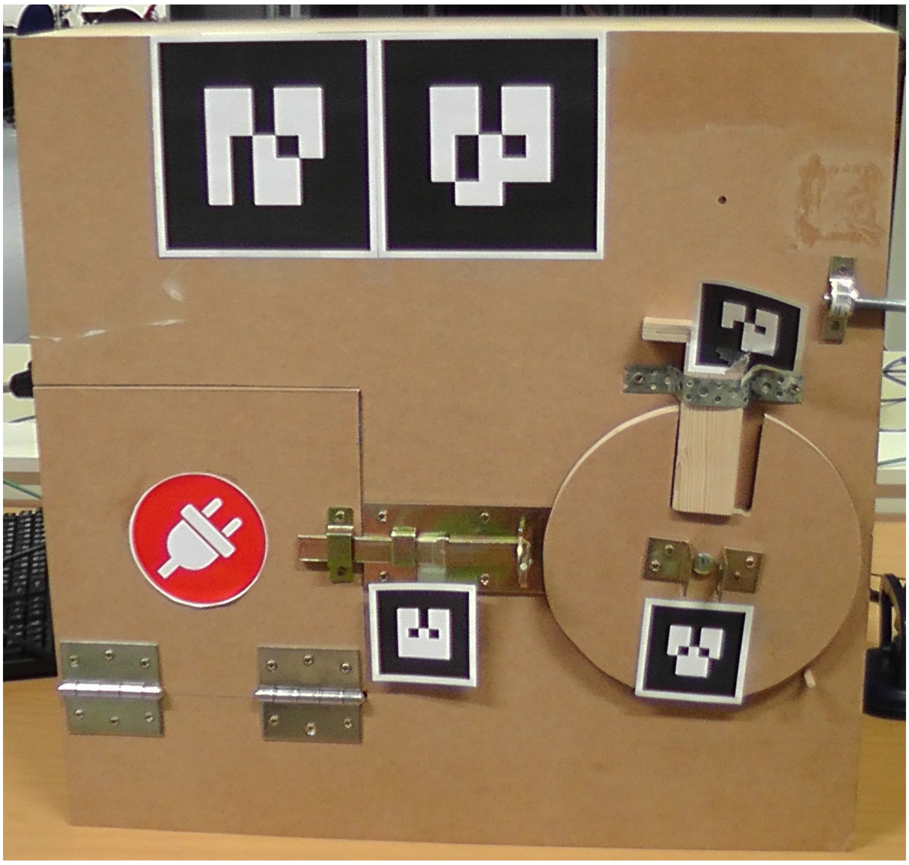

Figure 1.1: The lockbox I used to test the precision of the teleoperation.

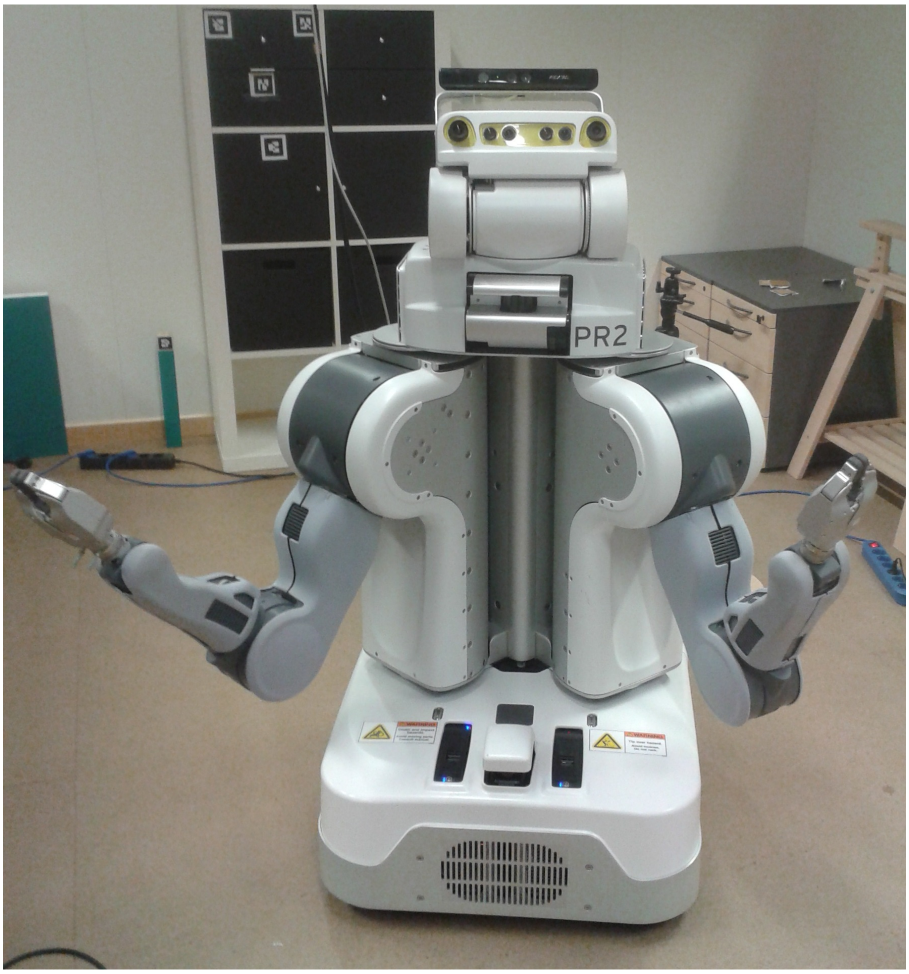

Figure 1.2: The PR2 which was used to perform the teleoperation.

algorithm to learn the robot the movements it needs to make to achieve the desired interaction.
Another way to make the robot interact with objects would be teleoperation. An intuitive way of doing teleoperation is tracking your hand and map the movement onto the robots endeffector. The data you get through teleoperation can also be used in machine learning algorithms to train the robot. A robot that that was trained with such an algorithm would be able to perform the interaction by itself.

In this work I improved an existing teleoperation. To test the quality of the teleoperation I used the lockbox shown in figure 1.1 as a benchmark. It features different movements and also rotation which makes it perfect for testing teleoperation. If the teleoperation is not precise enough to allow you to precisely move in one direction it is very hard to open the lockbox. All the teleoperation tasks where performed on a Willow Garage PR2, which can be seen in figure 1.2

Structure

This work is structured as follows: In Chapter 2 the related work is discussed, the different versions of teleoperation I tried are presented and discussed in Chapter 3. The evaluation of the final teleoperation version follows in Chapter 4. Problems that still remain are presented in Chapter 5. The final statement and future work are in Chapter 6.

2 Related work

Teleoperation is an important part of robotics. Robots are used for a wide variety of tasks, that humans either can't do because of physical limitations, like searching for people in a collapsed building after an earthquake, or it would be to dangerous for humans to perform this tasks, like defusing a bomb or working in a radioactive environment [LMI+11]. NASA and ESA use robots for space exploration, to gather information. This information is used to pick a landing site for a manned mission. Also the members of the crew are picked depending on the information, gathered by the robot[HK97].

Many of the teleoperation devices consist of a robot arm with seven degrees of freedom [RSE+14][HB12], which corresponds with the most accurate model for the human arm[PFBR01]. In order to get the best results with these teleoperation devices you need a robot that also has an arm with seven degrees of freedom. The teleoperation devices use motors with encoders to get the state of each joint, this information can then be mapped onto the robots arm. If the robot has force-torque sensors on every joint of the

Figure 2.1: The SAM exoskeleton used by the ESA.
Source: http://www.esa-telerobotics.net/gallery/Research/Haptic-robots-design-development/SAM-Exoskeleton/16 Date: May 14, 2016

arm, this information can be used to control the motors on the teleoperation device to let the user 'feel' the interaction with the object [RSE+14][HB12]. This technique can also be used to assist the user with certain predefined movements, like the devices surgeons use, do. In order to help the user make predefined movements, the forces the user feel along this path are reduced and the higher forces around this path guides the operator[TKC01].
For the use of teleoperation devices in space exploration it is important that these devices are as light as possible. When sending these devices into space to the astronauts every gram matters. It is expensive to send gear into space, for example it would cost $1654 to send a payload of 1kg into low earth orbit and $6617 to send it to Mars when using the Falcon Heavy[LLC]. In order to save money ESA constructed a very light teleoperation device, named SAM. This device consists of a exoskeleton for the right arm and a touchpad for the left arm. It weighs just 7.4kg and can be seen in figure 2.1 [LMI+11].
Another important consideration when developing a teleoperation device, is the delay between the teleoperation device and the controlled robot. When the time delay is not addressed in a force feedback teleoperation system, this can lead to instability. Also the loss of packages is important to consider. It is possible to make teleoperation delay-independent stable with the use of a transformation of the network signals, but the affects of package loss can only be minimized, they still have an impact on the stability of the teleoperation system [HMB09][HB04].

3 Discussion

3.1 Starting situation

The given teleoperation program used the Polhemus G4 position sensors attached to garden working gloves, the left glove with attached sensors can be seen in figure 3.1. This teleoperation gave you the ability to choose which endeffector you want to control, if you wanted you also could control both at the same time. When controlling both endeffectors you needed another person to start and stop the teleoperation for you, because every movement of the hands would also result in a movement of the robots endeffector. The position of the operators hand was directly mapped to the position of the robots endeffectors. The distance between the thumb and index finger were used to determine how much the gripper should open. When the teleoperation was started the robots endeffectors moved very fast into the position that the operators hand

Figure 3.1: The left glove with the Polhemus G4 position tracking system attached to it.

Figure 3.2: The gamepad used to initialize and exit the teleoperation.

dictated it. As the operator you had to guess where the origin of the calculation was, you had no way of knowing how far the robots endeffector would move. You also ran into issues when moving the endeffector down, because the table the whole setup was installed on, got in the way of the hand. Another problem was that you had no point of reference for your hands when operating. This resulted in unwanted movements, for example it was impossible to move straight up, you always ended up either slightly moving forwards, backwards or sideways. The inability to strictly move in the desired direction, together with the lack of any feedback on the currently used force by the robot, made it impossible to precisely interact with objects. This also lead to an inability to open the lockbox with this teleoperation.

3.2 Fixed movement restrictions

The first idea I had to improve the precision of the teleoperation, was to restrict the movement on certain axis. With the press of a button on the gamepad shown in 3.2 you could disable the movement on one axis. When the button got pressed a second time the movement was allowed again. You had the ability to disable the movement on the axis independent from each other, because every axis had a button assigned to it.
The buttons and the corresponding restrictions were:

- X : disable/enable movement on the x-axis
- Y : disable/enable movement on the y-axis
- B : disable/enable movement on the z-axis

- A : disable/enable rotation of the endeffector

In order to prevent unwanted movement, when re-enabling the movement, I calculated the difference of the operators hand position in every cycle. Only this difference in hand position was then used to calculate the movement of the endeffector. This usage of relative positions also enabled the operator to change his/her position without affecting the endeffector when the movement on all axis were disabled. That could be used to either get the hand into a more comfortable position, or move to a position which allowed the operator to get a better picture of the working area. When moving around you still had to stay near the G4 source.
The calculation of the rotation was absolute, like in the previous version, I could not think of any situation in which a relative calculation would be beneficial. A rotation of the endeffector that does not match the rotation of the operators hand would only make it more difficult to precisely get the desired rotation of the endeffector.
The ability to disable the movement on certain axis was a good idea and gave me the ability to move more precisely. A problem was the way I controlled the movement restrictions. Often I pressed the wrong button or could not remember which axis was disabled. Showing the disabled axis on the screen of the computer also did not help very much. I had to move my head in order to be able to see the information on the screen. The head movement often triggered a small but still noticeable movement of the hand which made the teleoperation difficult and not very satisfying to use.
The ability to disable the rotation was something I only used when re-positioning and I quickly stopped engaging the lock on the rotation at all. It was not difficult to remain the rotation while moving and this feature needed one of only four available buttons on the gamepad. The disabling of the rotation had also another problem. When re-enabling the rotation often the rotation of the operators hand did not match perfectly with the rotation of the endeffector. That made the endeffector rotate, and sometimes even move slightly because the robot needed to adjust its joint configuration in order to rotate the endeffector into the desired pose. The advantages that came with the ability to disable the rotation were not worth the trouble I had with the problems, so I did not carry this feature on to future versions.
I managed to open the lockbox partially because the robot and the lockbox were parallel. This allowed me to perform all of the required movements along the axis I could disable the movement on. When not being parallel the ability to disable the movement was useless, because the directions I had to move my hand did no longer match one of the axis. That meant I had to find a way to set the directions in which I wanted to restrict the movement flexible and more intuitive.

3.3 Force constraints

I realized that movement only happens in one direction at a time. This means that any modification of movement either needs to apply to the desired direction of movement or to every other direction. As a result of this realization I only needed one direction in order to set every restriction needed to make the teleoperation more precise. To get this direction I used the vector between the thumb and index finger of the left hand. The right hand was used to control the direction of the movement and also the rotation of the endeffector and the opening and closing of the gripper. The idea was to restrict the maximal force the robot uses and not the movement itself. In the desired movement direction I wanted to set the maximal allowed force near the maximum the robot is capable of. In every other direction the robot should use only the minimal force required to overcome gravity and friction. With this setup of forces the endeffector should be guided by external forces like movement restrictions of a joint of the lockbox.
The controller I used in the previous version was not capable of restricting the force, so I had to use another controller. This new controller behaved very different than the old one. It took much more time until the path of the endeffector to the desired location was computed, more than a second, this made it very difficult to move with high precision. It was also impossible to get the endeffector to rotate in the way wanted by the operator. Often it was just hanging down without moving at all. The problem probably occurred because the world model, this controller used, had no gravity. This affected the endeffector joint the most because all other joints had springs in it, that reduced the effect of the gravity. The reason was the only joint that did not have a spring that helped to overcome gravity. Another problem was that the endeffector started to move unpredictably when one joint was near its movement limit. The controller was tested and developed with a drawer that just needs movement in one direction. This has probably lead to a controller that has problems with more complex movements that also need a way to overcome the gravity. I also had the problem that the opening and closing of the gripper was not implemented into the controller.
All of this problems together made it impossible to get the teleoperation running while using this controller, so I had to search for another way to make the teleoperation more precise without being able to restrict the maximum force.

3.4 Flexible movement restrictions

I came up with the idea to use the best ideas of the previous teleoperation versions and fuse them together. The result was a teleoperation, that uses the active hand to dictate the movement and rotation of the endeffector, while the passive hand is used

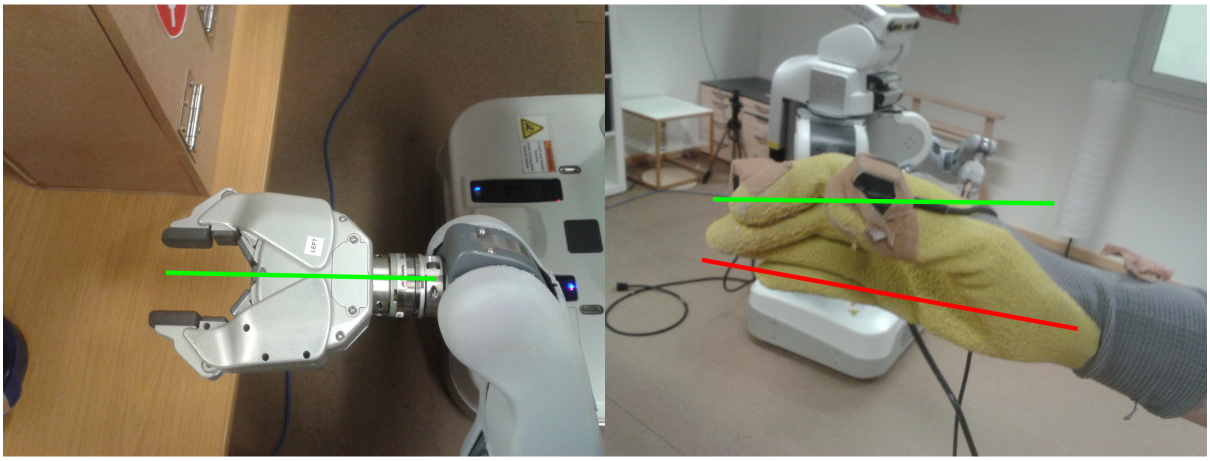

Figure 3.3: The axis of rotation on the robots gripper on the left and the axis of rotation on the human hand using the rotation of the elbow joint on the right.

to point into the direction in which the movement should be allowed. The length of the movement of the endeffector is calculated, using the length of the movement of the active hand. As the movement direction I use the vector between the thumb and index finger of the passive hand. This allows me to disable the movement restrictions entirely when the thumb and index finger of the passive hand touch. The disabling of the movement restrictions allows for quick movements towards objects without loosing the ability to have the high precision mode when needed. To ensure that the movement is intentional I compute the angle between the actual movement of the active hand and the direction of the movement restriction. Only when the angle is lower than 0.5 radian the endeffector moves.
This teleoperation also is ambidextrous, with the press of a button you can change not only the active and passive hand but also the endeffector you control. Another feature this teleoperation has, is the ability to home the joints of the robot. At the beginning of the teleoperation it is important to home the robots joints to ensure that the joint configuration ensures maximal movability, but it can also be needed when getting into a joint configuration that prohibits further movement.
I still use the gamepad to control the different modes and functions of the teleoperation. The buttons and their corresponding functions are:

- Y: Initialize the teleoperation, when holding down the button the robot homes itself
- X: Start the teleoperation. The controlled endeffector is the left one, the active hand is also the left one and the right one is passive.
- B: Start the teleoperation. The controlled endeffector is the right one, the active hand St also the right one and the left one is passive.

- A: Pause the teleoperation. To continue the teleoperation you have to press either X or B.

A problem I encountered with the rotation is, that the robots endeffector and the human hand has different axis of rotation when rotating with only one joint. The axis of rotation of the endeffector is exactly in the middle. A rotation of the human wrist is generated in the elbow, because of the offset of the thumb from the rotation axis of the Ulna and Radius, the vector between the thumb and the index finger also has an offset[WVV+05]. Figure 3.3 illustrates the different axis of rotation of the robot and the human hand. There is another way of rotating the fingers using the shoulder. For this rotation to only involve one joint, the middle point of the vector between the thumb and the index finger must be on the vector orthogonal to the rotation, otherwise other joints are involved to prevent a movement of the hand while rotating. This is not a problem when there is a point of reference near the hand, but because the teleoperation should be performed without any objects possibly hindering the operator in his/her movements the hand position of the operator tends to be unstable when using more than one joint for the rotation. Especially rotations towards the inside of the body are difficult to make with the rotation of the elbow. To ensure a proper rotation without also moving the endeffector it is best to restrict the movement while fully rotating, so that the only allowed direction of movement is forward. That way the up and down movement as well as the movement sideways that occurs when rotating the hand in a space without a point of reference does not affect the endeffector.

4 Evaluation

I tested how good the final version of the teleoperation worked with the help of some volunteers. The instructions I gave them were the following:

- To initialize the teleoperation you need to press the Y button on the keypad
- It is best to keep the Y button pressed until the robot is fully homed and stops moving
- After homing the robot you can either control the right endeffector when pressing the B button or the left endeffector when pressing the X button
- The A button pauses the teleoperation. After pausing the teleoperation you have to choose again which endeffector you want to control
- It is not recommended to start controlling the endeffector when there are obstacles near it. The initial movement in order to set the rotation of the endeffector to correspond to the rotation of the hand could result in unwanted touching of the obstacles
- Which hand is used to control the endeffector depends on the chosen mode. The right hand controls the right endeffector and the left hand controls the left endeffector.
- The rotation of the controlling hand is directly mapped onto the endeffector.
- The distance of the thumb and the index finger of the controlling hand corresponds to the amount the controlled gripper is opened.
- The hand that doesn't control the movement of the endeffector, is there to restrict its movement. When the thumb and index finger of the passive hand touches there is no movement restriction applied. Otherwise the vector between the thumb and index finger of the passive hand is used as the only possible movement direction.
- When the movement restriction is active the endeffector is only moved if the direction of movement loosely corresponds to the direction of allowed movement. This allows for re-positioning of the hand when moving orthogonal to the movement restriction

- Because of a different point of rotation of the robots endeffector and a human hand it is important to only allow movement forwards when fully rotating the endeffector. For slight adjustment of the rotation that is not needed.
- It is possible to get into joint configurations that do not allow any further movement. You will recognize when this happens. It is not possible to recover from such joint configurations. You have to pause the teleoperation with the A button and home the position with the Y button. Then you can start the teleoperation again
- To get the best joint configuration at the beginning you should not try to mimic the starting rotation of the endeffector. It is better to have the endeffector rotate at the beginning so that it is orthogonal to the robots base.

The restrictions of the movement are not very intuitive, most of the people who tried the teleoperation had problems getting used to it and tried opening the lockbox without them. All except for one person did not succeed in opening the lockbox without using the movement restrictions. Only after they realized the importance of the restriction and started to get used to using it, they succeeded in the opening of the locks. At the beginning I gave instructions in which direction they had to move in order to successfully open the locks. After they had opened the first two locks, which are the most difficult ones, I let them open the rest with nearly no instructions. Everyone managed to open the lockbox in less than 40 minutes. After they successfully managed to open the lockbox for the first time I gave them a break and then they tried to open it again. On the second try they got no help from me and they all where not only successful but also significantly faster. Everyone needed about seven to fifteen minutes to open the lockbox when trying it a second time. The ones that needed the most time had the most problems in giving accurate movement restrictions with their passive hand. Often the index finger was either too low or too to high when wanting to move parallel to the floor. This resulted in less accurate movements and therefore they needed more time to open the locks.
Everyone said they felt more confident and secure in their movements after they used the teleoperation for a while.

A problem on which everyone agreed upon, was the need for a better visual system. The robot often obstructed the line of sight to the working area and the cameras mounted on the robot can't be used to see the working area either. They are nearly always obstructed by the robots arm. The best solution to this would probably be a virtual reality headset, that allows you to see the robots arm transparent. This would give you a better view on the working area while still allowing you to see the current configuration of the joints.

It was not possible to re-position far enough to get a better view angle, because this would have resulted in a position that would have been to far away from the G4 source. When moving too far away from the G4 source the received position data either gets extremely noisy or you receive no data at all.

Another problem that was often mentioned, was the annoying recurrent failure of the G4 system. More of the problems with the G4 are in the chapter 5.
Everyone also agreed that feeling the force the robot currently uses and a recording and mapping of all joints and not only the hand position would have been better. That would also have solved the problem, that the robot gets into a joint configuration where it is impossible to move in the direction you want. You could simply let the operator 'feel' that he comes close to a limit of a joint. This would give the operator the possibility to re-position the joints in order to get into a joint configuration that allows for the wanted movement. Another possible improvement that was often managed, was to use the feet to control the different modes of the teleoperation instead of the gamepad. That way you have no need to re-position your hands and can not trigger any unwanted movement.

5 Problems

There are some problems that are fairly easy to solve, but also some fundamental problems in the way this teleoperation works. The easiest problem to solve is not really a problem, it is more a restriction. Because the robot preferred to move its base rather than moving its endeffector, I had to fix the base to prohibit the robot from turning when moving the endeffector sideways. As a result of fixing the base, the robot also does not move its base up in order to get the endeffector to higher positions. This results in a limited operation height of 1.35m.

Figure 5.1: The robot in a joint state which prohibits further movement.

One of the more fundamental problems is the use of the Polhemus G4 position sensor. It randomly either fails to retrieve any data at all, or the data received is incomplete and can not be used. When these failures occur it sometimes also sends noisy data, which makes it impossible to make any precise movement. I have not figured out any way to either predict when this failures might happen or solve this problem completely. When the G4 system fails you have to wait until it works again. This can take any amount of time, from just a few seconds to a few minutes.
Another problem with the G4 system is that there is no way to close the program in a way that does not crash the driver. To restart the driver you have to unplug the USB dongle and re-plug it. In order to not have to constantly unplug and re-plug the USB dongle and thereby creating unnecessary wear on my USB port and the USB dongle I outsourced the G4 data retrieval to another program which only receives the G4 data and sends it to the main teleoperation program. This allows me to close and start the teleoperation without having to deal with the issues of a driver crash when exiting the program. Rarely the G4 driver also crashes without any noticeably reason. A way to solve this issues is to use another system to get the position of the operators hand.
Another big problem is that you only control the position of the endeffector and the robot has to figure out a joint configuration that results in the desired endeffector position. That means you not only have to monitor the position of the endeffector but also the current joint configuration and have to adapt your movement accordingly. Otherwise it can happen that you end up in a joint configuration that does not allow the movement you want to make. The robot will try to follow your movement orders anyway and this will result in a position from which it is impossible to recover like the one that can be seen in figure 5.1. When this happens you have to stop the teleoperation, home the robot and start again. This is not only very difficult, but also annoying and requires knowledge of the way the robot will change its joint configuration with a specific change in endeffector position.

6 Conclusion and future work

The teleoperation in its current state is usable for tasks that can easily be repeated, or moved back into their original state without having any issues. This can be tasks like opening a drawer, a door or the lockbox. Another requirement that needs to be met in order to be able to perform the task, is that you need to have a good view on the working area from the position you are controlling the robot from. I would not recommend to use the teleoperation for tasks, in which one very little mistake can either have devastating effects, like when defusing a bomb, or can cost a lot of money and destroy objects.
The most important improvement I would suggest for future versions is the reliability of the hand tracking. When improving the hand tracking you could also start tracking the elbow and the shoulder joint. That way you would also get rid of the problem that the robot gets into an unrecoverable joint configuration. If doing so with a motor and an encoder you end up with a way to also give the user feedback on the force the robot currently uses in every joint, which would also make the restriction of movement unnecessary. This would also be the current state of the art way to solve the teleoperation, this solution is not only used in space exploration but also by surgeons [LMI+11][TKC01].
It would also be nice to find another solution for changing the different modes of the teleoperation, maybe the use of a device that can be controlled with the feet would be a better solution than the use of a gamepad.
Also there is the need for a visual system to enable the operator to get a better view on the working area. The upcoming virtual reality headsets HTC Vive and Oculus Rift may be a good option to give the operator an easy way to see the working area and also intuitively adjust the view angle when needed. I would recommend to show the arm only transparent, or at least give the user the ability to make the arm transparent when wearing a virtual reality headset. Otherwise it is likely that the arm of the robot is obstructing the view onto the working area.

Bibliography

[HB04] S. Hirche, M. Buss. “Packet loss effects in passive telepresence systems.” In: *Decision and Control, 2004. CDC. 43rd IEEE Conference on*. Vol. 4. IEEE. 2004, pp. 4010–4015 (cit. on p. 14).

[HB12] S. Hirche, M. Buss. “Human-oriented control for haptic teleoperation.” In: *Proceedings of the IEEE* 100.3 (2012), pp. 623–647 (cit. on pp. 13, 14).

[HK97] S. J. Hoffman, D. I. Kaplan. “Human exploration of Mars: the reference mission of the NASA Mars exploration study team.” In: (1997) (cit. on p. 13).

[HMB09] S. Hirche, T. Matiakis, M. Buss. “A distributed controller approach for delay-independent stability of networked control systems.” In: *Automatica* 45.8 (2009), pp. 1828–1836 (cit. on p. 14).

[LLC] M. LLC. *SpaceX rocket capabilities* (cit. on p. 14).

[LMI+11] P. Letier, E. Motard, M. Ilzkovitz, A. Preumont, J. Verschueren. “SAM: Portable haptic arm exoskeleton upgrade technologies and new applications fields.” In: *proc. of the 11th ESA Workshop on Advanced Space Technologies for Robotics and Automation, Noordwijk*. 2011 (cit. on pp. 13, 14, 27).

[PFBR01] R. Prokopenko, A. Frolov, E. Biryukova, A. Roby-Brami. “Assessment of the accuracy of a human arm model with seven degrees of freedom.” In: *Journal of biomechanics* 34.2 (2001), pp. 177–185 (cit. on p. 13).

[RSE+14] J. Rebelo, T. Sednaoui, E. B. den Exter, T. Krueger, A. Schiele. “Bilateral Robot Teleoperation: A Wearable Arm Exoskeleton Featuring an Intuitive User Interface.” In: *IEEE Robotics & Automation Magazine* 4.21 (2014), pp. 62–69 (cit. on pp. 13, 14).

[TKC01] N. Turro, O. Khatib, E. Coste-Maniere. “Haptically augmented teleoperation.” In: *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*. Vol. 1. IEEE. 2001, pp. 386–392 (cit. on pp. 14, 27).

[WVV+05] G. Wu, F. C. Van der Helm, H. D. Veeger, M. Makhsous, P. Van Roy, C. Anglin, J. Nagels, A. R. Karduna, K. McQuade, X. Wang, et al. "ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion—Part II: shoulder, elbow, wrist and hand." In: *Journal of biomechanics* 38.5 (2005), pp. 981–992 (cit. on p. 20).

All links were last followed on March 17, 2016.

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

place, date, signature