



Model Making and Haptic Controller Design
for the ReFlex TakkTile

K.P. (Koen) De Meyere

MSc Report

Committee:

Dr.ir. J.F. Broenink
Dr.ir. D. Dresscher
Dr.ir. R.G.K.M. Aarts

December 2018

049RAM2018
Robotics and Mechatronics
EE-Math-CS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Preface

This report marks the ending of my time as a System and Control student at the University of Twente. It took me a while to get to this point, but I couldn't be happier. In the six and a half years I've been in the Netherlands I learned a lot about my home country, and made a lot of new memories.

With the ending of my time as a student drawing near, it is only fitting to thank the people who have helped me get this far.

Thanks.

Well, that's it really. Anyone that got me this far will at least try to read the beginning of my paper, so this thanks is directed straight at you. I really appreciate what you guys have done for me and even though the constant nagging can be infuriating, it did motivate me to try harder. And some of you distracted me from my work through games or swimming, but that only kept me sane through this otherwise stressful period.

Koen Paul De Meyere
Enschede, December 2018

Summary

The i-Botics project is a joint operation between TNO and the University of Twente that aims to create tele-operated devices that can be controlled from a remote location. This project aims to create an intuitive interaction between a robot and its operator while remaining stable and safe for both parties. The robot in question consists of a four-wheel base (Segway RMP 50 Omni) and a robotic arm (Kuka LWR 4+), the arm supporting a robotic hand (ReFlex TakkTile) for remote manipulations. This assignment is a small segment of the larger project, focusing mainly on the robotic hand's interface between the user and the robot through the use of Haptic feedback.

The goal of this research is to create an intuitive yet effective Haptic Interface for a ReFlex TakkTile using an Omega 7 Haptic device. Building upon earlier research, this assignment will focus on reducing reaction time and implement teleoperations such that the hand can be controlled remotely. The investigation will include possible control schemes, degrees of freedom required to operate the hand effectively and mapping options for the sensors available. A model of the hand will be made as a result of the research and should provide a clearer insight in the hand's operations. The final product should be a tele-operated hand with an intuitive Haptic Interface between the user and the robot.

Contents

Preface	iii
1 Introduction	1
1.1 Context	1
1.2 Research Proposal	1
1.3 Related Works	1
1.4 Overview	2
2 Background	3
3 Analysis of the System	5
3.1 Analysis of the ReFlex TakkTile	5
3.2 Course of Action	10
4 Design	11
4.1 Model of the ReFlex TakkTile	11
4.2 Actuation	13
4.3 Parameter Derivation	15
4.4 Model Evaluation	20
5 Control	23
5.1 Common Architectures	23
5.2 Suggested Architecture	26
6 Conclusions	29
6.1 Conclusion	29
6.2 Future Work	29
Appendix A - Full Model	31
Appendix B - Custom Blocks	32
Bibliography	34

1 Introduction

1.1 Context

The rapid growth in automated manipulation has provided an efficient and more cost-efficient method than human being could ever achieve. Automation has revolutionized the industrial world where repetitive tasks are performed on a daily basis. Robots are less accident prone, replaceable and feature much less down time than a regular employee. But what if some tasks require more complex skills, or some kind of complex manipulation in an unknown environment? This is one of the topics the i-Botics research group tries to investigate: How to perform complex humane manipulations while maintaining the benefits of the disposable and robust nature of robots.

In order to achieve this, research has been done into the making of a haptic interface between a robotic platform and an operator. As the user operates the machine from a distance, they will receive physical feedback from the robot to increase environmental awareness. A robotic platform was created to act as prototype, featuring a computer mainframe, omni-directional wheels, an 7DOF and a robotic hand. This platform aims to demonstrate the features and limitations of haptic feedback through telecommunication. The robotic arm has a well established haptic interface (van Teeffelen (2018)) and the robotic hand features a rudimentary haptic interface (Lammers (2017)).

Whereas haptic interfaces are a well established topic, the introduction of telecommunication creates additional variables within the system that reduces the transparency of the system, potentially rendering the system unstable. In order to guarantee the safety of the operator and the robot itself, it is essential to implement a way to guarantee the stability of the system, while maintaining an adequate level of transparency.

1.2 Research Proposal

The robotic platform created by i-Botics features two key manipulation devices that will provide feedback to the user: a Kuka LWR 4+ arm and ReFlex TakkTile hand. Both devices are subject to similar limitations introduced by telecommunication, namely time delays. When trying to create a bilateral control algorithm, the time delays introduced by sending and receiving commands through any communication channel can reduce both the transparency and stability of a system.

The arm has been investigated thoroughly by K. van Teeffelen (van Teeffelen (2018)) but the ReFlex TakkTile only has a basic implementation of a haptic interface (Lammers (2017)). This report will aim to improve the work of B. Lammers by implementing more complex control algorithms that can guarantee the stability of a system.

As a result, the research objective is defined as followed:

"Realize a teleoperated haptic interface between the ReFlex TakkTile and an Omega 7 Haptic Device."

1.3 Related Works

Several fields of research are actively developing haptic technologies, but for different purposes. The medical field aims to create an intuitive learning tool and high precision apparatus. The gaming industry attempts to improve immersion by providing physical feedback to its players, which would in turn improve their reactions.

The medical field invested a lot of resource in the development of haptic interfaces, having developed medical instrumentation that provide artificial tactile feedback, created haptic sim-

ulators to provide physical experience and more (Kapoor (2014)). One of the key characteristics that led to the development of haptics in this field is due to the intuitive way it can teach the user how to use an object. The quick adoptability of haptic interfaces makes it ideal for a learning environments, and is also why i-Botics is interested in implementing it onto a robotic platform.

The gaming industry has also implemented haptic feedback in a majority of their products (Orozco (2012)). In an attempt to improve immersion, the gaming industry has developed multiple ways to provide physical feedback back to the player, like an actuated steering wheel or a joystick. This immersion is an important aspect of haptic control and can improve performance of non-pre-programmed actions.

The i-Botics group has done research into haptic interfaces as well, specializing in telecommunications and the implications thereof. K. van Teeffelen has developed a haptic interface between the user and a KUKA Lwr arm (van Teeffelen (2018)), based off of the work of M. Franken (Franken and MacChelli (2011)). They employ the concept of passivity in order to guarantee the stability of the robot arm. Other researchers have also investigated the effects of teleoperations on the transparency of a system and how their control architecture can affect the system's behavior (Hashtrudi-Zaad and Salcudean (2001)).

1.4 Overview

This report will consist of six chapters, including this introductory section as the first chapter. Chapter 2 will provide the necessary background information about the tools used in this report and chapter 3 will detail the capabilities of the ReFlex TakkTile. Chapter 4 will focus on the modeling process of the ReFlex TakkTile and Chapter 5 will provide the framework for future controllers to be used on the ReFlex TakkTile. The final chapter will sum up the findings of this report and evaluate the various instances that could have been improved for future recommendations. All relevant information that could not fit the narrative of this report will be found at the very end, in the appendices.

2 Background

In order to fully understand the scope of this project, it is important to know the tools used in its process. This chapter is dedicated to a short description of the various hardware and programs used throughout the report. This report focusses on the use of an Omega 7 to control a ReFlex TakkTile Device, controlled through the Robotic Operating System. The model of the hand will be made using bond graph theory and implemented in 20-Sim.



Figure 2.1: Omega 7 (left), ReFlex TakkTile (middle), ROS and 20-Sim (right)

Omega 7

The device on the user end is the Force Dimension Omega 7, a unique 7 DoF Haptic Device. It can read out the position of the controller in a Cartesian Coordinate frame and provide force feedback in any of its Degrees of Freedom. This thesis, however, will work primarily with the 7th DoF unique to this iteration of the Omega controllers, the active grasping extension. The active

Grasping Specifications	
Workspace	25mm
Forces	$\pm 8.0N$
Resolution	0.006mm

Table 2.1: Specifications of the Trigger Mechanism on the Omega 7

grasping extension is small button like input device that can determine the depth to which the button is pushed. It can provide a maximum of 8N as force feedback and has a resolution of $\pm 0.006mm$, which is about 0.024% of the total range of 25mm.

ReFlex TakkTile

The second device of importance is the Reflex TakkTile. The Reflex TakkTile is a product derived from a Yale research group and was commercialized by Right Hand Robotics. It is a three fingered robotic hand with nine pressure sensors on each finger. It also provides four servomotors (Dynamixel MX-28T) to It is an under actuated device its proximal joints, providing 5 actuated degrees-of-freedom, leaving the distal phalanges unactuated. This results in an underactuated device, providing the challenge of this assignment.

The exact measurements of each component are unknown and will have to be determined through experimentation, most notably the spring constant of the Urethane joint between the distal and proximal phalanges of each finger. These values are essential to the making of a model of the hand in order to full grasp the extent of the hand's operations.

Robotic Operating System (ROS)

It is important to understand that the control algorithm will be done through the Robotic Operating System (ROS), an open source meta-operating system for robotic devices (wiki (2018)). It provides a communication infrastructure and implements several different styles of communication. ROS provides a multitude of tools that supplement robotic control software and simplifies communication between devices. By running ROS on two terminals it is possible to communicate between two devices through a Wi-Fi connection, which is one of the primary attributes required for this thesis.

Bond Graphs and 20-sim

At the core of this report is the modeling of the ReFlex TakkTile, and this was done using Bond Graphs in 20-Sim. Bond Graph theory is a generalized modeling method that can combine various multiple disciplines using the concept that energy behavior is domain independent. This method of modeling was chosen as it allows for modular models to be implemented with ease and because resources were readily available for the model making process.

The model itself was subsequently made in 20-Sim, a program by Controllab (Controllab Products (2008)). It is a modeling program built on the principle of bond graphs, and as such it is ideal for this assignment.

3 Analysis of the System

This chapter will detail the different analyses performed during this research which ultimately led to the design of the model. It first discusses the requirements of the system at hand, followed by an analysis of the established control theories. The performance of the available equipment will then be tested to establish the limitations of the setup. These tests will later be incorporated into the model of the ReFlex TakkTile, and subsequently be used to justify the choice of controller.

3.0.1 Device Categorization

The situation at hand features two devices communicating through telecommunication channels, that will provide feedback to each other; a process called bilateral control. The haptic interface, also called the master device, sends information about the desired activity to the robot to be operated, also called the slave device. These devices are subsequently categorized by their ability to send position or force data to their counter-part as either admittance or impedance type devices.

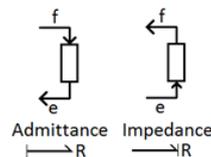


Figure 3.1: Energy flow in admittance vs. impedance type devices.

Admittance type devices will exert a force based on position data and are typically defined by their high back-drivability and low impedance (Hashtrudi-Zaad and Salcudean (2001)). Figure 3.1 shows that such an element could be represented as a resistive element. According to Hashtrudi-Zaad and Salcudean (2001), Admittance-Admittance type systems are prevalent in the industry of telerobotics and they go through great length to explore the stability of the different control architectures it can provide, which will be discussed in Section 5

Impedance type devices will move to a position based on a given force value and are typically characterized by their low back-drivability and low admittance. Figure 3.1 shows that their bond graph equivalent is a compliance element, reflecting this description. These types of devices are often used in industrial settings where robust actions have to be performed.

3.1 Analysis of the ReFlex TakkTile

Whereas the Omega 7 can easily be categorized as an Impedance type device, the ReFlex TakkTile sports several sensors and control possibilities, making it a polyvalent device suitable to multiple control configurations. Figure 3.2 depicts the location of the various sensors and devices with respect to the base of a finger and Table 3.1 lists the capabilities of said sensors.

	Control	Sensor
Motion	Velocity Control Position Control	Angular
Force	Motor Load Control	Motor Load Pressure (9×)

Table 3.1: Available control modes and sensors in the ReFlex TakkTile.

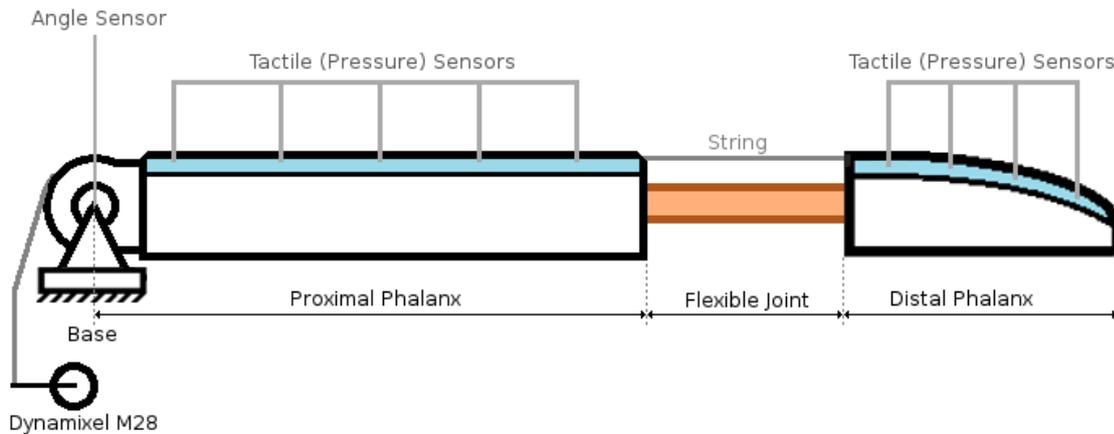


Figure 3.2: A diagram showing the location of the ReFlex TakkTile's sensor with respect to the base of a finger.

The Reflex TakkTile has an angular sensor in its base joint, allowing it to determine the position and velocity of this joint. It also contains a load sensor in the motor that calculates the force exerted based on inferred values, and several pressure sensors on each phalanx: five on the proximal phalanx and four on the distal phalanx of each finger. The performance and suitability of each sensor will be tested in order to determine whether the ReFlex TakkTile should be operated as an Admittance or an Impedance type device.

3.1.1 Position Controlled Behavior

The first test performed was to give the ReFlex TakkTile two different signals and to observe the data each of the sensors produced. Two motion profiles were generated: a sinusoidal wave and step responses. The sinusoidal wave would show how the ReFlex TakkTile reacts to a high frequency signal with small displacements, whereas the step responses were used to identify the ReFlex's behavior to abrupt motions. The data from both the angle sensor in the base and the force data in the motor were plotted in Figure 3.3. The pressure sensor data was omitted as the fingers did not come into contact with anything during their displacement and would not provide any insightful data.

The angle sensor is located in the base-proximal joint and measures the angle of the proximal phalanx with respect to its initial position. Once calibrated, the zero angle is determined as the initial position the finger is in when no load is applied.

As can be seen in Figure 3.3, there is an almost immediate reaction from the finger, but there is a clear offset between the desired value (blue line) and the measured value (red line). This offset appears to be ~ 0.25 radians, although this value was different from the sinusoidal input than the step signals. Looking into the code, it was revealed that the position controller uses the motor's orientation as controlled variable, and not the angle measured by the sensor. The controller uses this discrepancy to calculate the orientation of the distal phalanx mathematically. They create a rough estimate of the distal phalanx's orientation by subtracting the measured base-proximal from the desired set-point. The code itself admits that this could be improved, but it essentially gives a rudimentary method of identifying the unobservable domain.

The force sensor is located in the Dynamixel M28 itself, but is not a sensor in the true sense of the word. It calculates a force readout based on inferred values from the device, which means there is no actual force sensor within the motor. The force values given by the motor were tested in two ways: by giving a finger a position command and by giving a finger a force command.

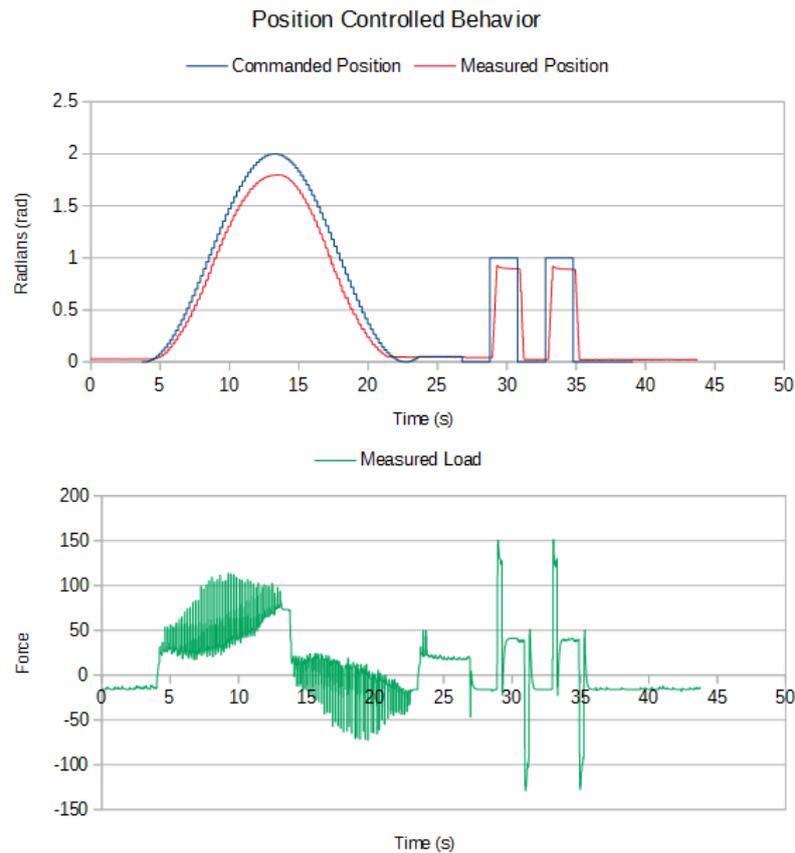


Figure 3.3: Response of the ReFlex TakkTile when given a position command.

The green plot in Figure 3.3 shows the force behavior as calculated by the Dynamixel M28. Whenever the finger received a new set-point, the force would spike to move the finger quickly. The plot reveals that these peaks are proportional to the slope of the slope of the set-point plot, or to the distance between the new set-point and the current position. Another noteworthy property of the plot is the range of forces measured by the motor. Whenever the set-point decreases, the measured load becomes negative, implying the string is acting a pushing force onto the motor. The scale of the values is also important to note, as the data suggests the motor is exerting a force of 100 or more Newton, which seems to be too big by a large margin.

The discrepancy between the angle sensor and the set-point is one of the major drawbacks of this control algorithm. The current implementation could be adapted to use the measured value, but this could potentially render the system less reactive. One of the benefits of using the motor orientation over the measured angle is that it allows the ReFlex TakkTile to grab an object using position control, allowing the finger to be controlled even if the proximal phalanx is obstructed. This control algorithm is not desirable for haptic implementations because the discrepancy between the angular sensor and the calculated angle already shows a significant off-set even though the proximal phalanx is not obstructed. No amount of calibrating would account for this off-set, making it an unreliable control algorithm in combination with the sensor. It should be possible to use the sensor's data instead of the calculated angle, which would increase the value of this controller.

3.1.2 Force Controlled Behavior

The second experiment performed was to give the ReFlex TakkTile a single force command to observe the behavior of the fingers. In Figure 3.3, the force measured when the finger is at 1

radian was equal to 38N, and thus applying a constant force of 38N should force the finger to reach 1 radian after a certain interval, depending on the amount of damping in the system.

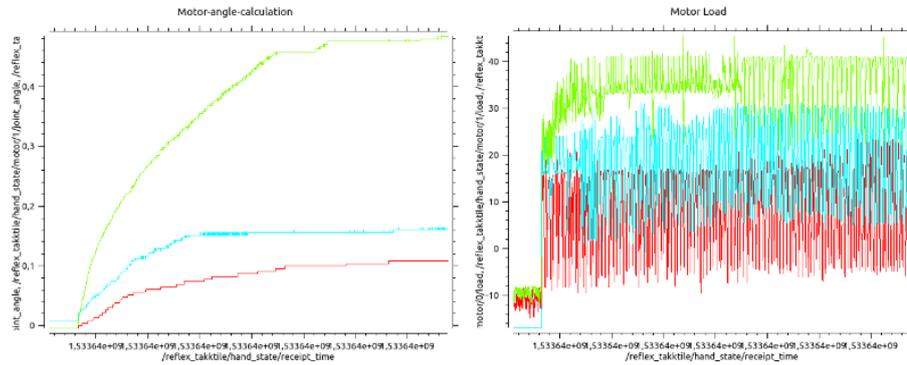


Figure 3.4: Response of the ReFlex TakkTile when given a Force command.

The data in Figure 3.4 shows data collected over a timespan of 5 minutes. In this period of time, the fingers - motor systems did not reach an equilibrium and were still moving when the test was halted. The finger subjected to a 38N force reach an angle of 0.489 radians, which is half of the expected equilibrium point. Another key characteristic of this experiment is that the force values show strong oscillations around the assigned force value, resulting in the average force being equivalent to the desired force. Although this oscillatory behavior appears to happen at quick intervals, the magnitude of the peaks can be as large as half the control force, which reduce the reliability of this control scheme. It should be possible to filter this behavior in order to use it reliably in a control algorithm.

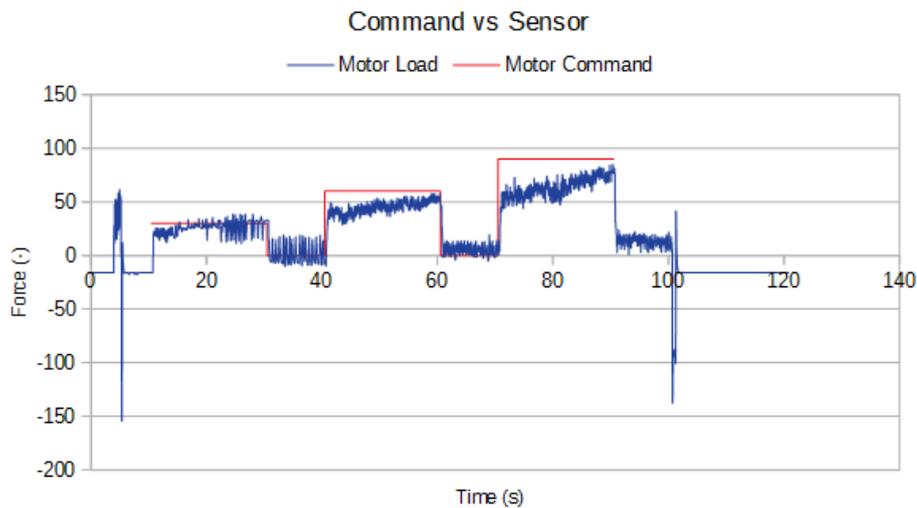


Figure 3.5: Response of the ReFlex TakkTile when given varying force commands.

Testing the motor even further, the motor was given a sequence of varying force commands to follow, in order to see how quick the device could adapt to new commands. Figure 3.5 shows that, at lower forces, the system is able to exert the desired force almost immediately, but at higher forces it requires time to reach the desired force. This is especially noticeable when the finger was given the command to exert a force of 90 units, as the initial force exerted was half of that. The measurements never reached the 90 units mark within the 20 second timeframe it was supposed.

The force functionalities of the ReFlex TakkTile are not optimal for most control schemes. Although it is possible to filter the oscillatory signals received from the force sensor, the time

required to achieve the desired force, especially larger forces, reduces the reliability of data as a whole.

3.1.3 Tactile Sensors

Whereas previous tests would compare known values with the read-out values, it is impossible to determine or set the pressure applied to each pressure sensor without knowing the properties of the material covering them. Instead, the ReFlex TakkTile was made to hold a cylindrical object in order to see how reliable the pressure sensors are at sensing obstacles. The set-point was set within the cylindrical object and the robotic hand was controlled using position control.

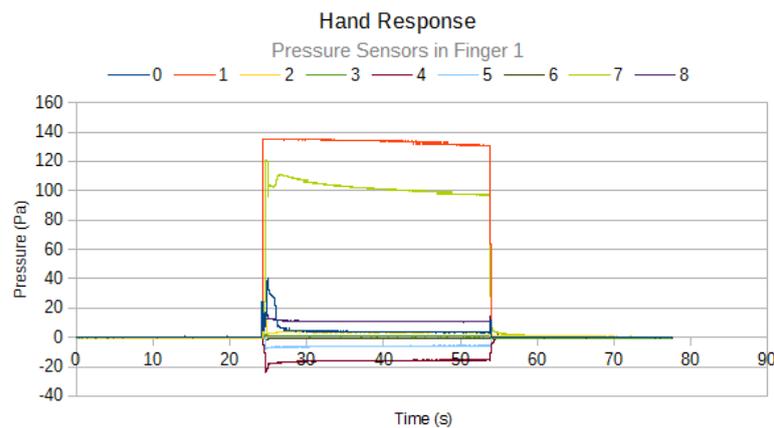


Figure 3.6: Data from Pressure sensors as ReFlex TakkTile grabs a cylindrical object.

The data in Figure 3.6 shows that the pressure values were consistently 0 until contact was made. Two of the nine pressure reported significant changes in pressure, while some reported minor changes. The reason for this is that the ReFlex TakkTile cannot make full contact with a cylindrical body, and it should not be expected for it to do so in the future. The large values are in direct contact with the test object, whereas the smaller increases are most likely due a distribution of load throughout the finger's surface material. The negative pressure value can be the result of these same deformation but due to its location, it actually decreases the pressure.

3.1.4 Other Noteworthy Behavior

Delays

The ReFlex TakkTile has a variety of delays built within its design, both intentionally and unintentionally. A safety mechanism was added to the hand's driver which induces a 0.09s delay between the reception of a command and its execution. This delay is present in order to prevent the Dynamixel motors from becoming unresponsive.

The effect of reducing the hard-coded delays was tested and the results indicated that the delay is indeed necessary. Using Bart Lammers's code (Lammers (2017)), the hand was connect to the Omega 7 and it froze up almost immediately. This was repeated several times and it became apparent that the forced delays are essential to the control of the hand. The value for this delay is arbitrary, as stated by Right Hand Robotics (Robotics (2014)) and there is no theoretical support as to why this is the optimal value.

Force Compensation

The ReFlex TakkTile has a built in overload compensation function that automatically triggers when the load on the motors exceeds a certain amount (Robotics (2014)). This reduces the risk of the hand destroying itself.

where the force seen in the motors fluctuate greatly. The ROS-log indicated that the force compensation activated repeatedly as the hand was grabbing an object, resulting in periodic behavior. This oscillation is not equal to either the publishing rate of the hand state (40Hz) or the controller (10Hz), therefore it is hypothesized that some form of aliasing occurs if the hand receives commands faster than it can process.

Another key aspect of this situation is the reduced publishing rate of the hand state. The state of the hand is constantly published at a rate of 40Hz until the hand grabs an object. The publishing rate then lowers to 28Hz which is a drastic drop in data sent out. It is hypothesized that this drop occurs due to the increased calculation load on the hand induced by the force compensation mentioned earlier.

3.2 Course of Action

Given the results in this chapter, it is impossible to determine which sensors are most suitable for a haptic controller. The position controller has a non-negligible off-set in its values and the force controller provides noisy signals. A bond graph model of the ReFlex TakkTile will be made in the next chapter to quantitatively judge which sensors and built in controllers are optimal in the case of a teleoperated haptic device.

4 Design

This chapter will detail the model making procedure, detailing the tests done to derive the values of each element and analyzing the performance of the final product. The first step was to identify the various elements within the ReFlex TakkTile, to quantify them and subsequently incorporate them into a Bond Graph Model. Section 4.1 will detail the identification process of the body and section 4.2 will document the modeling process of the actuation within the system. This is followed by the parameterization process in Section 4.3, all of which will then be evaluated on their approximation to the real finger in section 4.4.

4.1 Model of the ReFlex TakkTile

In order to make an appropriate controller for the system, a model was made using bond-graphs and screw theory. By using six dimensional bond elements it should be possible to simulate the 6 domains of screw theory, and as such a bond graph model was made in 20-sim, a simulation program based on bond graph theory (Controllab Products (2008)). The finger was first simplified into a model of ideal physical elements, based off of which a 2D bond-graph model was made. This model was then expanded to 3D using D. Dresscher's work on modular screw theory elements (Dresscher and Stramigioli (2010)).

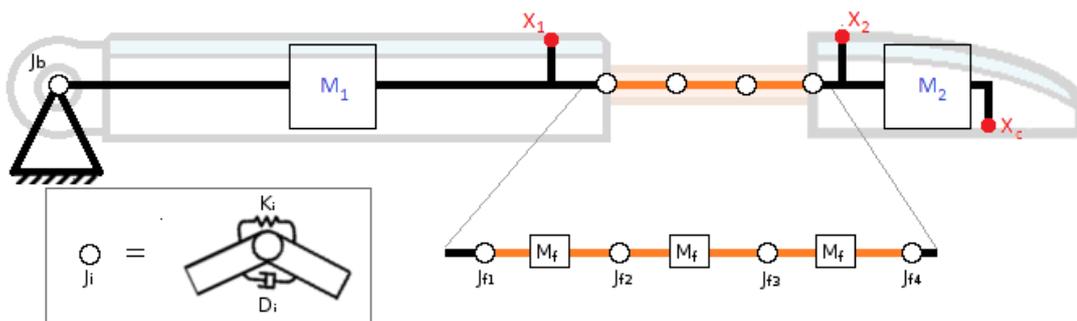


Figure 4.1: IPM model that depicts the various components of the ReFlex TakkTile.

Using the finger shown in 3.2, an Ideal Physical Model (IPM) was made using primarily standard elements, resulting in the model shown in figure 4.1. Using the base as reference point, all other elements will be expressed in relation to the frame of this component. As the base is a non-moving element, its inertial properties are neglected. The other elements do move, and thus the position of their respective centers of gravity should be included in the model. Both the proximal and distal phalanges are considered to be rigid bodies with their respective centers of gravity M_1 and M_2 in the locations as indicated by Figure 4.2. The model also distinguishes four points of interest: X_b , X_1 , X_2 and X_c . These are the positions at which the actuating string exerts a force on the finger and thus are essential position for when actuation is implemented in 4.2.

This model features two joints, of which one is a composition of several mass-spring elements in succession. The latter is meant to approximate the flexible nature of the poly-urethane connection as a crude Finite Element Model. It was decided to use four joints as approximation in order to keep calculation times fast while still simulating the desired behavior. Each joint will feature a spring and a damping constant based on the friction, spring values and internal values of the materials used.

To transcribe this model into bond graphs, the points of common efforts and common flows were identified and noted down as 0 and 1 junctions. The relevant inertial, resistive and ca-

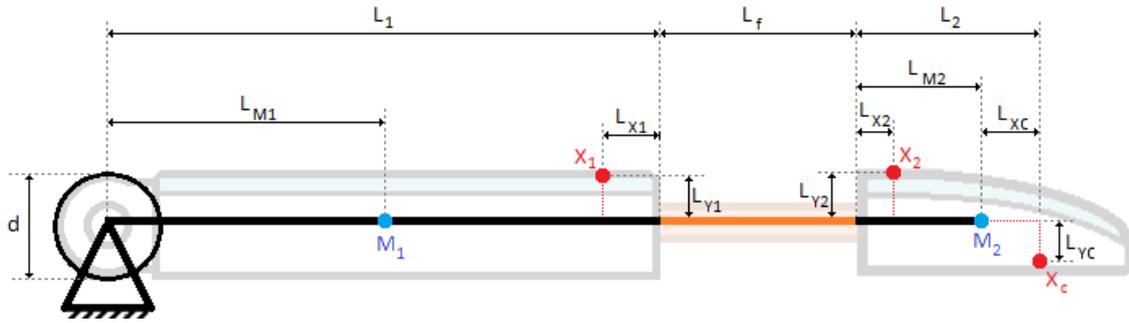


Figure 4.2: Diagram showing all relevant length definitions within the ReFlex TakkTile.

capacitive elements were subsequently added to depict the masses, damping and springs within the system, with each element working in both the rotational and translational domains. An inertial element therefore contains the mass and inertia values of the relevant body, a resistive element contains both rotational and translational damping and a capacitive element contain both torsional and translational spring values. These values were left empty during the modeling phase until Section 4.3.

This model was then adapted to work calculate each elements' reactions within their own reference frame, using D. Dresscher's work (Dresscher and Stramigioli (2010)). A transformation matrix was added between every 1 and 0 junction to change the frame in which the bond graph is operating. This induces unwanted gyroscopic effects into the system, which is why new element called the Eulerian Junction Structure was added (Stramigioli (1999)).

The behavior at the relevant positions, namely points X_b , X_1 , X_2 and X_c , are determined using adjoint H matrices corresponding to each position. Points X_1 , X_2 and X_c are fixed to specific parts of the system, and by calculating their frames as a function of their respective parts, the transformation matrices will remain constant. The same does not apply to X_b , however, as this

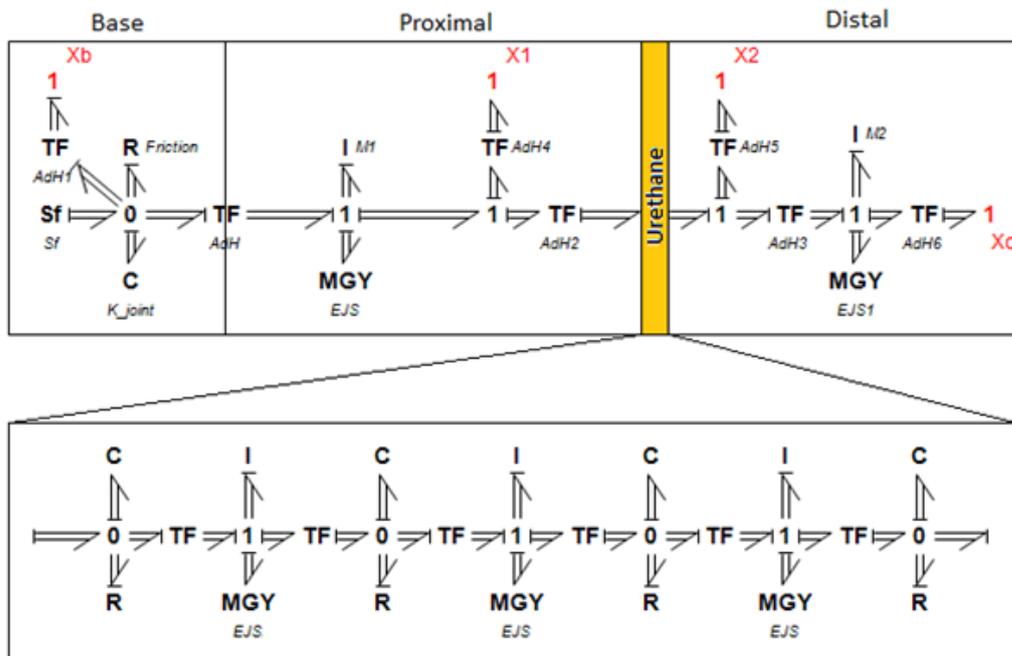


Figure 4.3: Bond graph model using 2D screw theory.

point changed with the orientation of the base joint. The transformation matrices for points X_1 and X_b are given in Equation 4.2.

$$H_0^{x_1} = H_0^\phi(\phi(t))H_\phi^i H_i^{x_1} \quad H_0^b = H_0^{\frac{\phi}{2}}(\phi(t))H_{\frac{\phi}{2}}^b \quad (4.1)$$

$$H_{trans} = \begin{bmatrix} 1 & 0 & 0 & x_i \\ 0 & 1 & 0 & y_i \\ 0 & 0 & 1 & z_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad H_{rot}(\phi_i) = \begin{bmatrix} \cos(\phi_i) & 0 & \sin(\phi_i) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\phi_i) & 0 & \cos(\phi_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

Where H_{trans} represents a translational frame change and H_{rot} represents a rotational frame change. The equation for X_1 calculates the frame of the base as a function to this rotation θ first, and subsequently derives the center of gravity of the proximal phalanx using the second term H_θ^i . This is essential as the reference frame of the body is already calculated for its inertial properties. The other two points, X_2 and X_c are derived in a similar way, using the H matrices of their respective body fixed frames. The equation for X_b is different because it relies on half the angle of the base joint, but is always a set distance r_b away from the pivot point. There are no inertial properties attached to any frame this point is derived from, a property that indicates that its influence on the system is minimal.

4.2 Actuation

The model derived thus far consists solely of the finger as this part of the device can easily be represented using screw theory. The element that actuates the device, namely the string attached to the distal phalanx, is a non-rigid body that is impossible to model using screw theory without resorting to infinitely small elements. The string can exert a force on any point the body makes contact with it and change its orientation. After some initial testing, the string acts mostly inelastic and, for the sake of simplicity, was assumed to be thus. With this assumption, the tension in the string can be considered constant between points of contact.

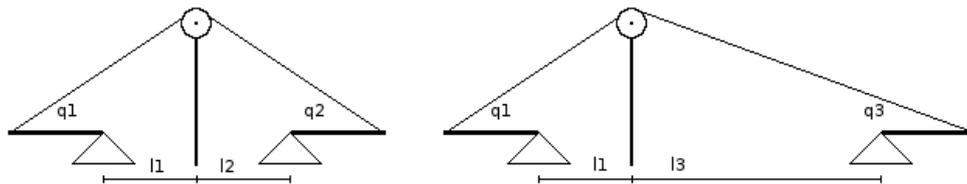


Figure 4.4: IPM model to show influence of geometry on string interactions.

In an ideal scenario, the string will act as a transformer between two systems, directly applying forces from one system to another. Figure 4.4 is meant to show how geometry affects the way a string translates forces to another system. It depicts two systems where the left side is geometrically the same, whereas the right side has the secondary system slightly further from the string's pivot point. The torque applied to the system is dependent on the string's direction. The ReFlex TakkTile features several instances where the string's direction changes and thus requires extensive knowledge of the system's states. In the model presented in this paper, all positions and orientations of the string are calculated and thus can be used to derive the direction of the force it applies on the various points of contact, which is an essential property of a string.

The next step was to analyze the forces exerted at the position X_b , X_1 , X_2 and X_c , as indicated by Figure 4.1. These are the points at which the string comes in contact with the finger, and as

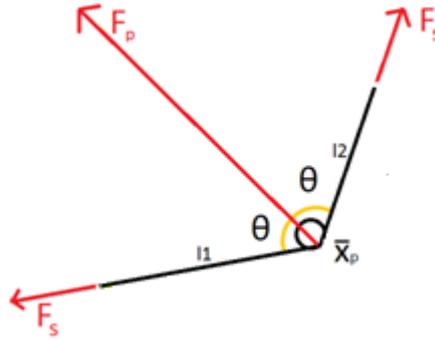


Figure 4.5: Orientation and forces within a string/finger connection.

such is able to exert a force on it. Figure 4.5 shows how a string can exert a force on a trolley without being attached to it, which boils down to Equation 4.3.

$$F_p = 2F_s \cos(\theta) \quad (4.3)$$

In this equation, the force exerted on the pulley is dependent on angle θ , the deflection of the string. If this angle is larger than or equal to 90deg, the string does not actually exert a force on the pulley. Ideally, this behavior would be incorporated into the model using a transformer element, but the velocity of the string is not solely dependent of the movement of the pulley, but the geometry surrounding it. The velocity of the string would be a function of $l1$ and $l2$, as indicated in Figure 4.5, which are in turn dependent on other elements of the ReFlex TakkTile. It is impossible to create a self contained transformer that will calculate the force and velocity exchange between the pivot point and the string.

As a way to circumvent this problem, several attempts were made to implement the behavior but the final implementation was to use effort and flow sources at the identified points of interest.

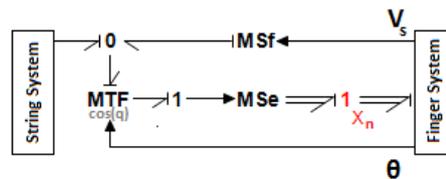


Figure 4.6: Bond graph model of the pseudo transformer.

The submodel shown in Figure 4.6 represents a transformer exerting a force on a point using screw theory, henceforth called a Pseudo-Transformer. The element presented is not reversible and exerts forces at three different locations on the finger. This element is also not power continuous, which is why this element cannot be considered a traditional transformer element. The discontinuity is due to the fact that a string cannot store or transfer any torques or rotational motions. Any rotational data is lost at the point of contact, and while it might be possible to model such behavior using rigid bodies for both the body and the string, this approach is ill suited to the string's constraints as it can only apply a force in one direction.

One of the requirements for this element is to calculate the overall velocity of the string (V_s). Figure 4.7 shows the various positions and lengths that are crucial to the string's movement. The string is strung between points X_m and X_c , and comes in contact at the finger at points X_1 and X_2 and all along the base between X_{b1} and X_{b2} . The length of the string can be calculated by taking the distance between each sequential point of contact, and differentiating the sum.

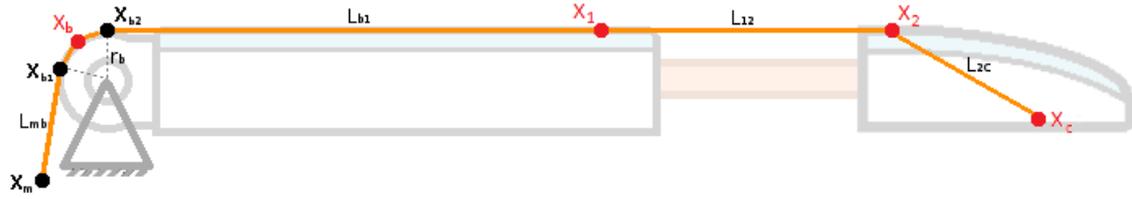


Figure 4.7: IPM model that depicts the various components of the ReFlex TakkTile.

Several distances, namely the L_{mb} , L_{b1} and L_{2c} distances are constant.

$$\begin{aligned}
 l_s &= r_b \frac{\phi + \alpha}{\pi} + l_{12} + l_{mb} + l_{b1} + l_{2c}, \quad r_b = 0.5d \\
 v_s &= r_b \dot{\phi}(t) + \dot{l}_{12}(t), \\
 l_{12}(t) &= \sqrt{(x_2(t) - x_1(t))^2 + (y_2(t) - y_1(t))^2 + (z_2(t) - z_1(t))^2}
 \end{aligned} \tag{4.4}$$

The equations shown in 4.4 show how interconnected the string's velocity is with multiple parts of the finger. This supports the claim that it is impossible to create localized transformers that translates the finger's motion into the string's velocity. The velocity is calculated within the model, and then fed to the flow source as shown in Figure 4.6.

The effort source in Figure 4.6 is based off of Equation 4.3. Using the θ_i values calculated externally, this effort source should also keep in mind the direction in which the force is applied. This is relatively simple for all points of interest as they each have their own reference frame. Equation 4.5 shows the effort values exerted at the various points of interest.

$$\bar{T}_1(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -2 \cos(\theta_{1,1}(t)) \cos(\theta_1(t)) F_s \\ 0 \\ 2 \cos(\theta_{1,1}(t)) \sin(\theta_1(t)) F_s \end{bmatrix} \quad \bar{T}_2(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -2 \cos(\theta_{2,2}(t)) \cos(\theta_2(t)) F_s \\ 0 \\ 2 \cos(\theta_{2,2}(t)) \sin(\theta_2(t)) F_s \end{bmatrix} \quad \bar{T}_c = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -2 \cos(q) F_s \\ 0 \\ 2 \sin(q) F_s \end{bmatrix} \tag{4.5}$$

Where $\theta_{1,1}$ and $\theta_{2,2}$ are the angles of the reaction force θ_n in the proximal and distal frame respectively. Although initially implemented, the effort source acting on X_b was removed because the force exerted does not impact the rest of the system. As a result, the final implementation of the pseudo-transformer features one flow source acting on the string's sub-model and three effort sources acting on the finger's three points of interest. This element is not power continuous but due to the nature of the string, it does not have to be.

4.3 Parameter Derivation

Although the bond graph model has been devised, the values of the various components need to be determined. The center of mass, the inertial properties of the various bodies and the spring constants in the various joints are all unknown properties that are required for the model to work. Most of the dimensions shown in Figure 4.2 were readily available, with the exception of the center of mass of each element. In order to find these values, a SolidWorks model was made and given the material properties of the material each component is made of.

The model shown in Figure 4.8 uses elements obtained from the RightHandRobotics GIT repository (GIT), which were subsequently modified to better represent the finger at hand. Analysis of each individual component yielded a mass and inertias in all 3 frames, which are required in the final model of the finger. The flexible joint was not available on the GIT repository, and as



Figure 4.8: SolidWorks model of the ReFlex TakkTile used to derive inertial properties.

such was characterized manually. The polyurethane material has a variable density depending on the casting method, but because the exact method of production is unknown, the average density was used. Using the density and volume of the flexible joint, the average mass of the joint and its inertial values were calculated. The other difference between the SolidWorks model and the actual finger is the presence of metallic pins at points X_1 and X_2 . The mass of these pins are negligible compared to the mass of the other component, and therefore left out.

Although the inertial properties were obtained, there are two instances where springs act on the finger: the torsional spring at the base joint and the flexible joint joining the proximal and distal phalanges. There was no documentation on either component and thus the values had to be determined manually. The base joint was best characterized experimentally as there is no data on the spring used, whereas the flexible joint could potentially be characterized based on the materials it is made of. Due to plastic deformations, however, the flexible joint is unlikely to behave as an ideal poly-urethane joint would. As such, this joint was also characterized through experiments.

$$\tau_t = L_1 \sin(\theta) = K\theta = 0 \quad (4.6)$$

For the spring in the base joint, a known force was applied at a set distance resulting in a change in angle that remains constant at equilibrium. This angle was measured using a protractor and using the formula in Equation 4.6, the torsional spring constant was calculated. The constant force was applied by hanging a weight of 50 g at the end of the proximal phalanx, resulting in a small change in angle in the base joint. The spring value was determined to be 5.47 Nm/rad.

The torsional joint was tested in the same way, by fixing the proximal phalanx and hanging a mass from the end of the distal phalanx. A 20g mass was used to procure a small deflection, which yielded a spring value of 5.928 Nm/rad. As the flexible joint is to be modeled as four torsional springs in series, this value was divided by 4m, resulting in a spring constant of 1.482 Nm/rad for each spring in the model of the joint.

Motor Properties

The motor had shown signs of being a non-ideal effort source (see Section 3.1.2), and thus needed to be fully analyzed in order to incorporate any irregularities. The two main characteristics of the motor were tested: the internal friction of the device and input-output ratio of given force signals.

The first test performed was to determine the effect of friction on the system. The motor was made to pull on an unloaded string and as such should reach a terminal velocity due to friction eventually. Thus, by measuring the angular position of the motor as a function of time, the

slope of the friction dominated response would be equivalent to the damping coefficient of the system.

$$F = M\ddot{x} + D\dot{x} + kx, \quad M = 0, \quad K = 0 \quad (4.7)$$

The data in Figure 4.9 was obtained by pulling at the finger when it was at 1.8 radians, whereas the data measures the position of the motor, thus once the motor position surpassed the 1.8 radians mark, it started to pull on the finger itself. It resulted in the data shown in Table 4.9.

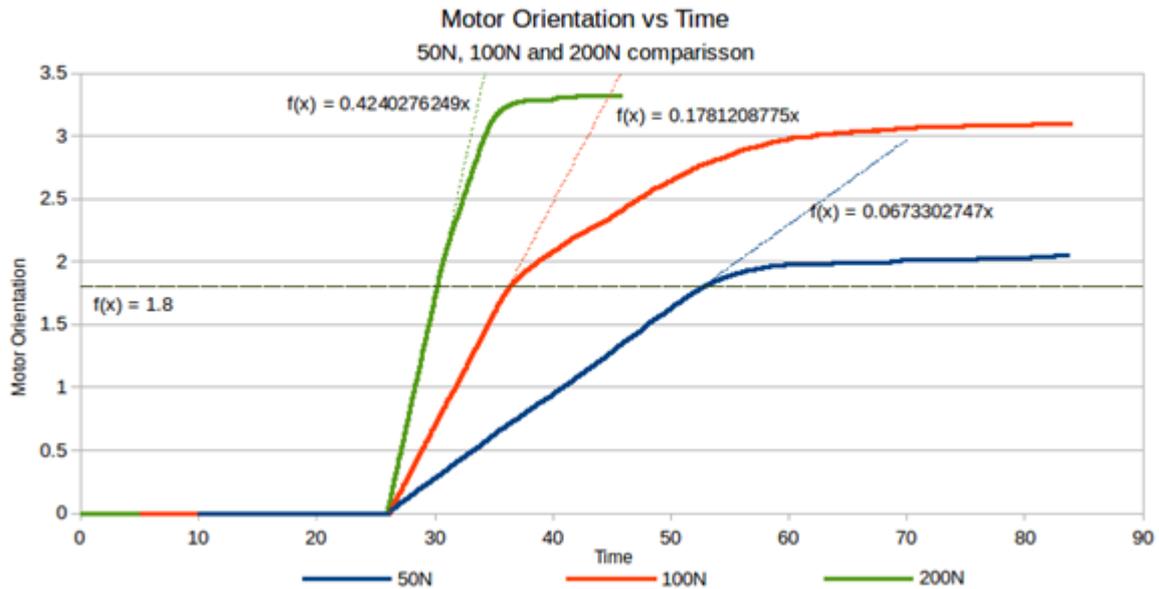


Figure 4.9: Pulling experiment on an unloaded string using 50 units (blue), 100 units (red) and 200 units (green).

One of the key characteristics of this data is that it reaches a terminal velocity almost instantly, suggesting that the effects of damping are prominently present in motor itself. It improves the analysis of the slope because there are more data points to calculate the slope with. These slopes were then used to calculate the friction value as a linear function of the input values.

<i>Motor input</i>	<i>50units</i>	<i>100units</i>	<i>200units</i>
Slope	0.06733	0.1781	0.4240
Damping Value	742.6	561.5	471.7

Table 4.1: Experimental data by applying weights to the ReFlex TakkTile string and comparing them to the simulated values.

The units for these values would theoretically be Newton seconds per radians, but the input values are not the force values exerted by the motor and thus no meaningful units can be attributed to the damping values. This relationship was subsequently implemented into a Coulomb-Viscous friction model provided by 20-sim (Controllab Products (2008)), but this caused the simulation time to increase drastically, and thus the friction function was adjusted slightly.

The second test performed was to attach the motor to a spring and measure its elongation when given a certain force. When the motor and spring reach an equilibrium, the acceleration and velocity values turn to 0, and thus only the spring has an influence on the system's position.

The aim is to determine the force exerted by the motor for any given command.

$$F = M\ddot{x} + D\dot{x} + Kx, \quad \ddot{x} = \dot{x} = 0 \quad (4.8)$$

By knowing the value of the spring's constant, it is possible to determine the force exerted (see Equation 4.8). The spring constant in this case was 500 N/m and the spring was hung above the motor to normalize the effects of gravity on the system. The motor was made to pull the spring with no initial elongation until the spring did not elongate any further. The spring's elongation was measured when the spring stopped moving, thus measuring the force exerted by the motor at that point.

The data obtained from this experiment was not accurate enough to be able to make any conclusive observations. For the motor-spring system to reach an equilibrium it would generally take several minutes, but even then the motor would still occasionally move further. The highest force inputs also required the longest stabilization time, some of them needing more than an hour to reach an observable equilibrium. As the spring was only extending over time, a second set of data was obtained where the system approach the equilibrium from a much higher initial tension and, when released, the spring would contract over time. This was done to increase the accuracy of the data obtained, as the values of the first set had a much higher chance of being too low.

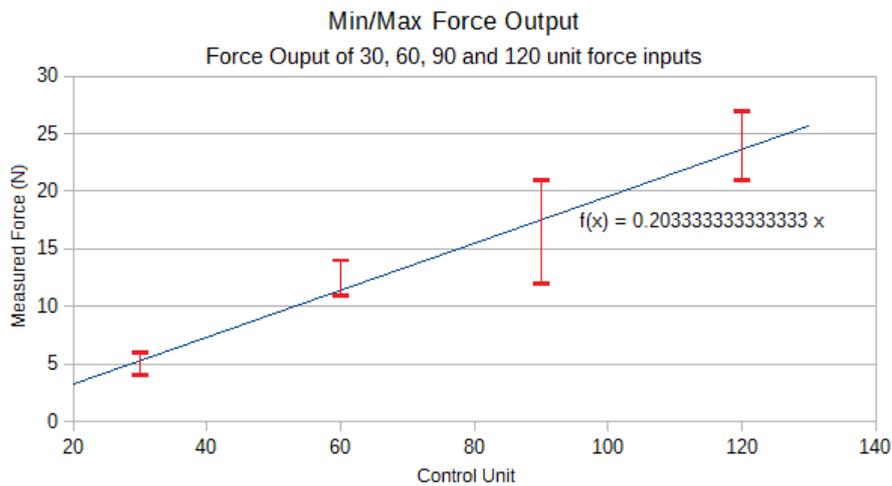


Figure 4.10: Force outputs measured as a function of the motor's input value.

The range between the lower bound and maximum value increases as the strength of the input command increased. The time required to reach an equilibrium took substantially longer as well and thus it is hypothesized that the increase in variance is due to them not reaching full equilibrium. Without indication of it being non-linear, a best fit was applied in order to determine a common factor. The line in Figure 4.10 reveals that the relationship between the input value is ~5 times the output value. Note that the data ranges for each input value are too large for this to be a conclusive result, but due to time constraints this was taken as the main mapping ratio.

With the friction and motor ratio identified, it is possible to create a bond graph model of the motor, and by extension, the string. Figure 4.11 features the bond graph that models the one dimensional behavior of the string. In this model, the friction is approximated using the Rf element, while the motor ratio block converts input signals into force units. This was subsequently implemented into the full model of the finger. The force exerted by this sub-system was used to calculate the forces acting on points X_1 , X_2 and X_c using the elements described

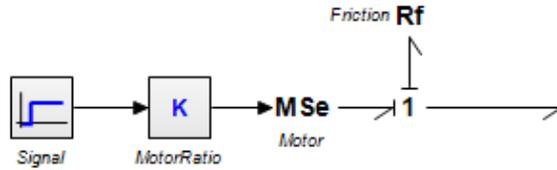


Figure 4.11: Bond graph of the Motor Element.

in Section 4.2. This force is derived from a zero junction connecting the motor submodel to the pseudo-transformer's flow source.

The final implementation uses the modular screw elements made by D. Dresscher (Dresscher and Stramigioli (2010)) where each joint and body and separated into submodels. The four submodels (Base, Joint1, Link_proximal, Urethane and Link_distal) all have two primary input and 2 primary output signals, with the exception of Joint1 who also exports the angle of the base joint. The model was made to mirror the composition as shown in Figure 4.1, with points X_b , X_1 , X_2 and X_c highlighted in red. Using the position of each body element in the origin frame given by these H-matrices, the StateCalc block was able to calculate the reaction angles as shown in Figure 4.5 and the length l_2 .

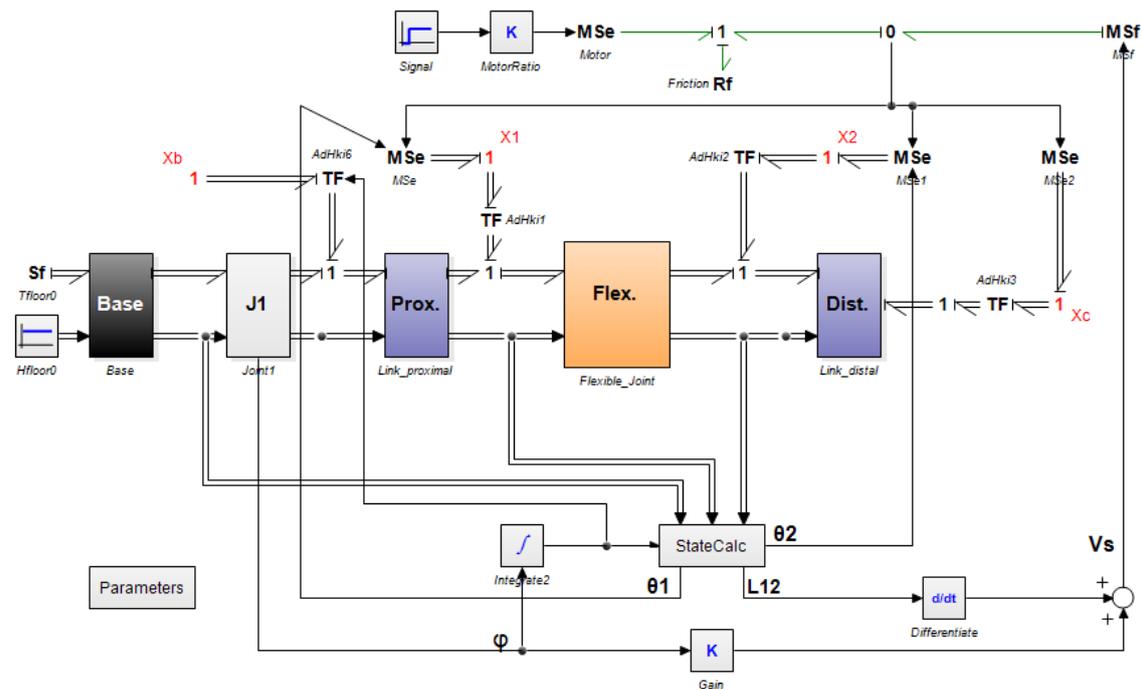


Figure 4.12: Bond Graph Model using the Geometrical method.

This model shows similar behavior to the ReFlex TaktTile, but that is an qualitative observation. The next step is to test the validity and accuracy of this model, which will be discussed in Section 4.4.

4.4 Model Evaluation

In order to test the validity of the model, several tests were performed and compared to the simulated equivalent. The first test was to apply a force directly onto the finger to compare the finger and string's model with actual values. Then, the finger was subjected to a constant force exerted by the Dynamixel MX28 to validate the incorporated motor model, and the last test is to actuate the finger using a block wave in order to test the dynamic properties of the model.

Constant Force - Weight

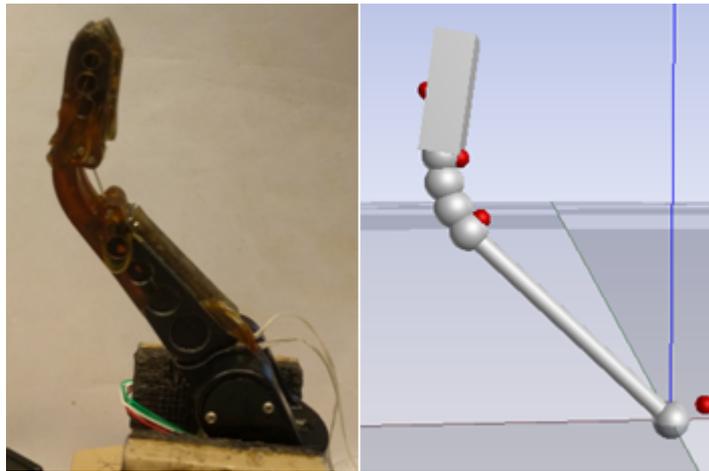


Figure 4.13: Comparison between the physical finger and the model subject to a 9.81N force.

A force of 9.81N was applied to the string by attaching a 1kg weight at the end and letting gravity pull it down unhindered. Figure 4.13 compares the physical finger with the model and they both take on the same shape. The model in the right features red indicators to show the points of contact. Besides giving a nice visual confirmation, this simple test is limited by the fact that the finger in question is not attached to the ReFlex TakkTile's main body, thus unable to use the sensors present in the system. The angles were measured using a protractor, and a force of 9.81N resulted in an angle of 0.8 radians (rounded to 1 decimal point due to the inaccuracy of using a protractor). The simulation, however, resulted in an angle of 0.76 radians.

<i>ForceApplied</i>	<i>PhysicalFinger</i>	<i>Simulation</i>
15.7N (1500g)	~2.35	2.407
9.8N (1000g)	~0.785	0.767
6.9N (700g)	~0.087	0
4.9N (500g)	0	0

Table 4.2: Experimental data by applying weights to the ReFlex TakkTile string and comparing them to the simulated values.

This data shows that the model of the finger and the string are able to calculate the position of the finger with an accuracy of 0.087 radians. This discrepancy is caused by the plastic deformation in the flexible joint, as can be seen in Figure 4.14. Plastic deformation can cause the flexible joint to have different spring values due to changes to the density and structure of the material. Although these values were derived experimentally, further deformations can be caused by extensive use of the ReFlex TakkTile, reducing the accuracy of the model.

To further test the validity of the model, the sensors of the device had to be used, but in order to do so the finger had to be attached to the motor. Because of this, the following test will use the ReFlex TakkTile's motor as actuation device with as goal to validate the dynamics of the model.

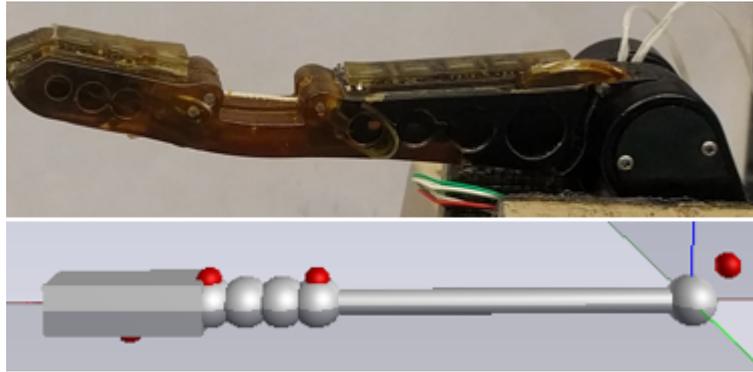


Figure 4.14: Comparison between the physical finger and the model subject to a 0N force, showing the plastic deformation.

Constant Force - Motor

The second test performed was to apply a constant force using the motor. The motor was given the command to exert a constant force in its own units, and the base-proximal joint angle was measured. Based on the experiments done in Section 4.3, the system is expected to have a slow response to the force input due to the internal damping in the system. This experiment aims to validate the motor model, specifically the unit conversion and the internal damping.

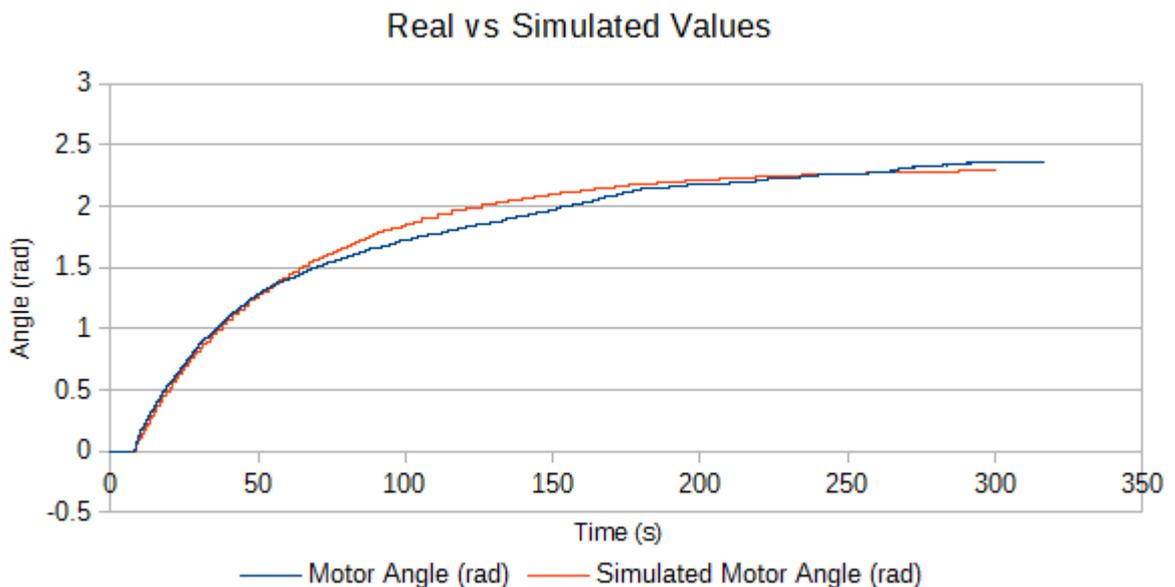


Figure 4.15: Comparison between the physical finger and the model subject to a 60 units force.

The plot in Figure 4.15 shows the motor's orientation as a function of time when given a 60-unit command. As can be seen, the initial behavior appears to be identical, but as the finger approaches equilibrium, the effects of friction appear to impact the finger's motion more than the simulation estimates. Friction is a difficult element to model and it is hypothesized that this discrepancy is caused by the friction element applied. The Coulomb Viscous friction model is a relatively ideal element while the friction within the finger

Another noteworthy characteristic is that the Motor Angle surpasses the simulation again after having slowed down to friction, behavior that suggests that the motor ratio might be inaccurate. The trend line in Figure 4.10 was not a tight fit, meaning there could be some variation in the

ratio calculated. Adjusting this ratio also requires some re-adjustment to the damping values, which could potentially improve the friction behavior discussed previously.

Variable Force - Motor

The last test performed as to see how the motor would react to a varying force command. To do this, an the virtual finger was subjected to a series of forces (30,60 and 90 units) and made to react to them for 20 seconds each. These values have shown the most observable deviations within the virtual finger, and the step size was kept significantly large as to show a visible change within the results.

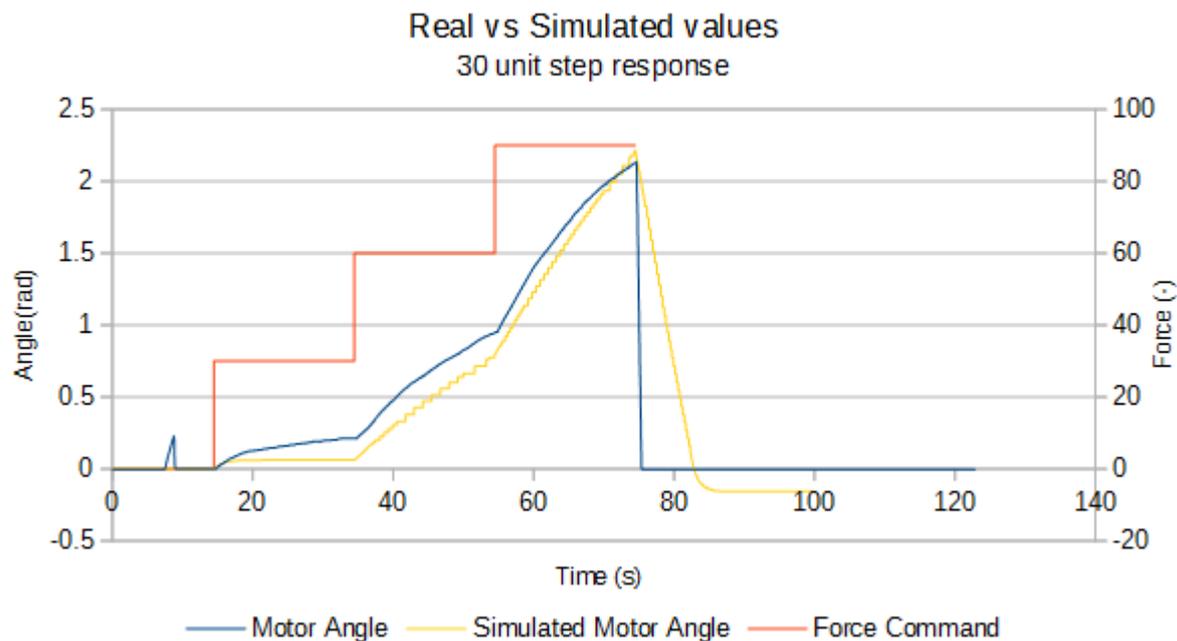


Figure 4.16: Comparison between the physical finger and the model subject to an block ramp signal.

The data in Figure 4.16 shows that the simulation and real finger behave similarly in both scenarios, although there is a slight offset within the data. It is hypothesized that this offset is due to the plastic deformation shown in Figure 4.14, which would cause the smaller lower force ranges to behave differently. This difference becomes less as the rate of deformation within the finger decreases with strong forces. The simulation data does intersect with the experimental data once the finger is subjected to a 90 units force. This too is most likely cause by the friction element.

Once the signals ended, the finger was told to return to the initial condition using a position command. This is where the simulation and experimental data differ quite substantially. The bond graph model does not feature position control, and as such the behavior was approximated by using the force values the finger read out when it returned to the original position. This resulted in the delayed reaction after the signal ends. This could be caused by several reason, but it is hypothesized that the friction at this point becomes non-existent, as position controlled commands have shown no signs of internal damping.

The behavior of the model is sufficiently close to the behavior of the physical finger. The model provides useful insight in the working of the ReFlex TakkTile and would help in controller development in the future. One of the most important observations to make is that Dynamixel motors are notoriously bad at handling force commands and readouts, and that the finger is best not operated using this as actuator. Several attempts were made to model the Dynamixel's behavior, but the model does not simulate it perfectly.

5 Control

A working model has been established in Chapter 4, based off of which a controller is to be made. Due to time constraints, this report will detail the potential controllers available to the set-up and evaluate their applicability. The following chapter will first lay the foundation for the various controllers available for both teleoperated and haptic devices, and subsequently present a system overview of how the controller would be best implemented.

First, the different control architectures will be discussed. The theories below will expand on the Admittance and Impedance type devices presented in Chapter 3.0.1 and how these concepts are applied in modern control algorithms. The Omega 7 was established as an impedance type device and the ReFlex TakkTile could be operated as both an admittance or impedance type device. After the tests conducted in Chapter 3, the performance of the force control was shown to be unreliable, supported by the model made in Chapter 4. As a result, the ReFlex TakkTile will preferentially be operated as an Impedance type device, but both options will be discussed.

5.1 Common Architectures

Considering the ReFlex TakkTile as a potential Impedance or Admittance type device, there are 3 types of control architectures available: Direct control, Impedance Control and a hybrid Impedance-Admittance Control. Each controller will be described and evaluated based on their value to the situation at hand.

Direct Control

Direct Control is achieved when the master and slave are operating in opposing causalities. This type of control would be applicable when the Omega 7 is operated as an Impedance and the ReFlex TakkTile is operated as an Admittance type device. This is also the approach taken by Bart Lammer (Lammers (2017)), who mapped the position of the Omega 7 trigger to the angular position of the ReFlex TakkTile, and the pressure data from the ReFlex TakkTile to the force exerted by the trigger.



Figure 5.1: A bond graph representation of Direct Control.

This approach, albeit practical in design, would result in a single closed-loop feedback system, which turns unstable when time delays are introduced. It is recommended to use a low pass filter (Hashtrudi-Zaad and Salcudean (2001)) for such a situation, or other concepts to increase the stability of such a system. The biggest drawback is that the force exerted by the hand cannot be controlled directly, thus making it hard to control the strength of the ReFlex TakkTile’s grasp. This was one of the desired features of the controller, which this control architecture would not support.

Impedance Control (P-F Control)

The second type of controller is the impedance controller that is generally used on Impedance-Impedance systems. The Impedance controller effectively changes the causality of a controller by introducing a virtual spring in the loop, as can be seen in Figure 5.2. This type of controller was implemented by Kees van Teeffelen (van Teeffelen (2018)) connecting the Omega 7 to the

KUKA LWR through a virtual spring. The main benefit of such a design is that it maps the user's motions to the motions of the slave device. This improves the transparency of the system as it gives every motion a single outcome, unlike direct control where several different actions could result in the same reaction.

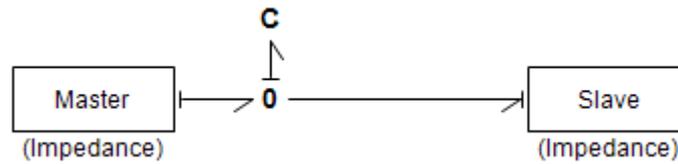


Figure 5.2: A bond graph representation of Impedance Control.

This type of controller is common in teleoperations and has been proven to work using passivity control (van Teeffelen (2018)). It does require the ReFlex TakkTile to operate as an Impedance type device, which it is ill suited to do. This report has thoroughly investigated the capabilities of the motor and the force control is too unreliable for such an application. As such, Impedance Control alone is not a viable option.

Hybrid Controller

The last type of controller that is viable is to use a hybrid controller, combining the virtual spring induced through the impedance controller with a desired plant system. The main goal for this type of controller is to apply the established Impedance controller to ReFlex TakkTile operating as an Admittance type device. A hybrid controller is presented by Ott et al. (Ott and Nakamura (2010)), and using their definition of an admittance controller it is possible to describe it as shown in Figure 5.3. Unlike the controller suggested in the paper, however, it should be possible to apply the impedance controller to the Omega 7 and the admittance controller to the ReFlex TakkTile.

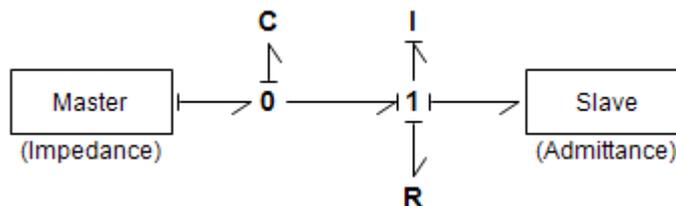


Figure 5.3: A bond graph representation of Impedance-Admittance Control.

The main benefit of using this type of controller is that it is possible to use an established energy passive controller as part of its core, but the side effect is that it might induce unwanted behavior into the system. It is a roundabout way to achieve the same results as Direct Control and there is no literature on the topic, thus putting into question the stability of such a system.

In the end, the Impedance Controller would be ideal if the Force commands were reliable and the Hybrid Controller does not have enough literature to support its stability, leaving the Direct Controller. Direct Control is generally unstable when time delays are introduced and thus alternative methods of guaranteeing stability need to be investigated.

5.1.1 Energy Based Control

One of the potential ways to determine stability is to analyse a system through an energy based analysis, an approach often used in haptic control Colgate and Schenkel (1997). The transparency of the system is dictated by the the various components of the device, the optimization of which is ideal for a passivity based design.

The use of this approach has the potential to be influenced by time delays (Lawrence (1988)), which can often lead to instabilities. To deal for this, the passivity theorem Hannaford and Ryu (2002) can be applied to guarantee stability of the system.

Passivity and Time Delays

The concept of passivity is defined as a system where the integral of the power extracted over time does not exceed the initial energy stored in the system. (Weir and Colgate (2008)) This means that, by limiting the movement and force exerted by a device, its stability can be guaranteed. Coupling such a device to another passive system would remain passive, thereby remaining stable. This allows for the creation of modular segments, which in turn promotes reusability and a more stable working environment.

Time delays lead to active behavior in otherwise passive parts of the system. If the total amount of generated energy becomes larger than the amount of dissipated energy, the entire system becomes active and unstable. This is unfavorable in any scenario, a situation that can be avoided through extensive energy management. Each motion contains a set amount of energy, and thus by communicating this maximum energy to the recipient device, it should be possible to impose limiters on their potential movement, thus guaranteeing stability.

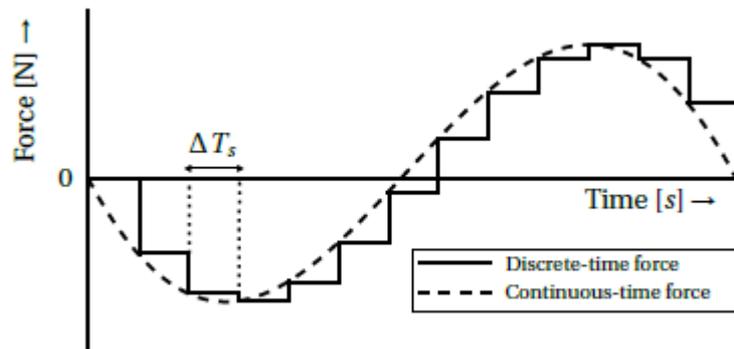


Figure 5.4: Discretization of a Force Signal (van Teeffelen (2018)).

Discretization is another issue that can affect the passivity of a system. As can be seen in Figure 5.4, the discretization of a signal causes the device to exert a force for longer than expected, creating a smaller time delay in the system. Albeit smaller in scale than communication delays, it is a consistent change in the energy stored in the system. In the case of repetitive motion/force signals, however, the energy stored can be minimized over time.

Guaranteeing Passivity

Due to the induction of time-delays, passivity cannot be guaranteed. In order to address the issue of fluctuating energy, the energy of the motion should be dissipated if it exceeds the internal energy of the system, the lack of energy being ignored as it does not change the passivity of the system. A variety of approaches are discussed by Franken et al. (2011), detailing the benefits and problems of each approach, of which two are especially relevant to this situation: The Passive Set-Position Modulation approach by Lee et al. (2010) and their own proposed Dual Layer Framework.

The Passive Set-Position Modulation Framework is centered around a spring-damper controller. Due to the nature of discrete signals, the sudden jumps in the set-point would create large jumps in energy within the virtual spring used to control the device. The energy exerted is subsequently limited by a concept called an energy tank, a layer that determines the possible energy a system can exert based on how much it has stored. The energy tank contains the virtual damper and is used to dissipate any excess energy. This approach has shown successful

implementation Lee and Huang (2010) but according to Franken makes assumptions about the continuity of the system that are inherently faulty. The assumption that the control system can be regarded as a continuous system and disregards the effects of discretization, a procedure known to generate energy Franken and MacChelli (2011).

The Dual Layer Framework expands the PSPM Framework by separating it into two algorithms: the Transparency Layer and the Passivity Layer. The Transparency Layer calculates the optimal outputs of the system for a desired action which maximizes the transparency of the system. The Passivity Layer is used to dissipate the surplus of energy induced by time delays by adding a virtual damper to the desired motion. The benefit of this approach is that the choice of controller will not affect the stability or transparency of the system, allowing for the choice of non-passive techniques such as signal filtering.

Of the two methods described, the Dual Layer Framework has shown good results within the i-Botics research group and is more suitable to a modular implementation. The ReFlex TakkTile has shown that, when commanded to exert a specific force, it features large noise peaks in its execution, which add energy to the system. The noise could be filtered out, but the difference between force commanded and force exerted would still remain an issue. Using the Dual Layer Framework it should be possible to fine-tune the controller to take these impulses into account without it influencing the transparency of the system.

5.2 Suggested Architecture

Having established the various controllers available, the final architecture that is recommended is to use the ReFlex TakkTile as an Impedance type device and to use Franken's passivity theory to guarantee stability of the system. Although the force sensor in the motor is inadequate for the types of applications envisioned, the pressure sensors on each finger would still be able to yield reliable information about its surroundings. This structure had been attempted before by B. Lammers (Lammers (2017)) but it lacked passivity control. His implementation also averaged the pressure measurements for each finger, which is not an efficient way to process the data since not all tactile sensors are used when interacting with an object, as shown in Section 3.1.3. It is advised to use the maximum pressure value per finger instead as it allows the Omega 7 to react to a wider range of values, thus increasing the transparency of the system.

This would imply that the ReFlex TakkTile would subsequently be controlled using velocity or position commands. Data in Chapter 3 revealed that the position control of the ReFlex TakkTile is suitable to the desired tasks at hand.

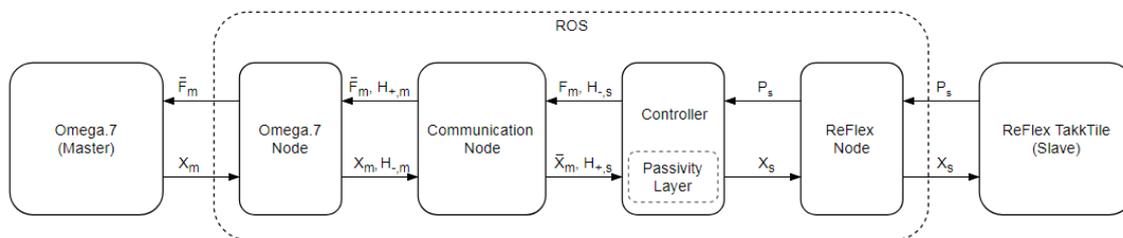


Figure 5.5: recommended architecture based off of van Teeffelen's work (van Teeffelen (2018)).

The architecture shown in Figure 5.5 is based off of van Teeffelen's work, applying his foundation to the ReFlex TakkTile. The principal idea is to operate the entire system using a central processing unit, the Intel NUC. The Omega 7 node would be modified in order to send energy packages through the communication node, which would subsequently be used to limit the controller on the NUC's end. The controller would receive the delayed position data of the Omega 7 \bar{X}_m and the pressure values P of the ReFlex TakkTile. In return, it would send out a force

command F_m towards the Master device and a position command to the ReFlex. The force command would subsequently be bundled together with an energy value $H_{+,s}$ which would be used to limit the reactions on the Omega 7's side.

This architecture is best suited to the i-Botics platform considering the limitations of the ReFlex TakkTile and the available hardware. Energy based control guarantees the passivity of the system and thus its stability, which is essential in teleoperated setups. The main issue with the ReFlex TakkTile is that it is an under observed and an unactuated device, which could hamper the transparency of the system.

6 Conclusions

6.1 Conclusion

Considering the research objective proposed in Section 1.2:

"Realize a teleoperated haptic interface between the ReFlex TakkTile and an Omega 7 Haptic Device."

The full scope of this objective has not been achieved, but considerable advancements have been made for future development. A bond graph model presented in Chapter 4, which was subsequently evaluated through experimentation and shown to behave similar to the physical system. One particular component of interest is the pseudo-transformer, an element created just for this project as the string does not behave as a rigid body. It is not a power continuous element, a key characteristic of Bond Graph elements, but the experiments have shown that the model approximates reality to a satisfying degree featuring this element.

Extensive research was done on the functionality of the Dynamixel M28, the sole actuator within the ReFlex TakkTile. Although the motor is very adept at performing position and velocity based commands, its force commands leave something to be desired. Several options were considered, namely to filter the force signals received from the motor, or to use the pressure sensors present on each finger. The latter was chosen as the results shown in Chapter 3 show that the force functionality of the Dynamixel simply does not provide adequate control over the system.

A control architecture was also recommended in Chapter 5, using the findings of this report as motivation for the design choices. It incorporates the works of previous i-Botics works and presents a passive teleoperated haptic controller using one main controller. It is recommended to operate the Omega 7 as an Admittance type device, while operating the ReFlex TakkTile as an Impedance type device.

6.2 Future Work

Although the initial goal was to create and implement a haptic controller for the ReFlex TakkTile, too much time was spent on the creation of a model that this ended up the final product of this research. Based on this work, the following actions are recommended:

- Implement the controller as suggest in Chapter 5.
- It is recommended to try and work with the ReFlex TakkTile 2 over the current version, as it features an IMU in its distal phalanx, making the system fully observable. Although it is not essential to the creation of a haptic interface, the extra sensor would greatly enhance controller's performance, improving the transparency of the system in general.
- It is possible to try to actuate each finger individually using a different controller. As research is being done into control mechanisms that do not use the Omega 7, the trigger functionality would have to be replaced.
- Form a modeling perspective, it could be interesting to further investigates of non-fixed pulley systems on a string. Although this report created a pseudo-transformer for the sake of simplicity, this element is only applicable to the ReFlex TakkTile. It should be possible to create a standardized function for a moveable pulley on a string.

Appendix A - Full Model

A break down of the bond graph model used for simulations.

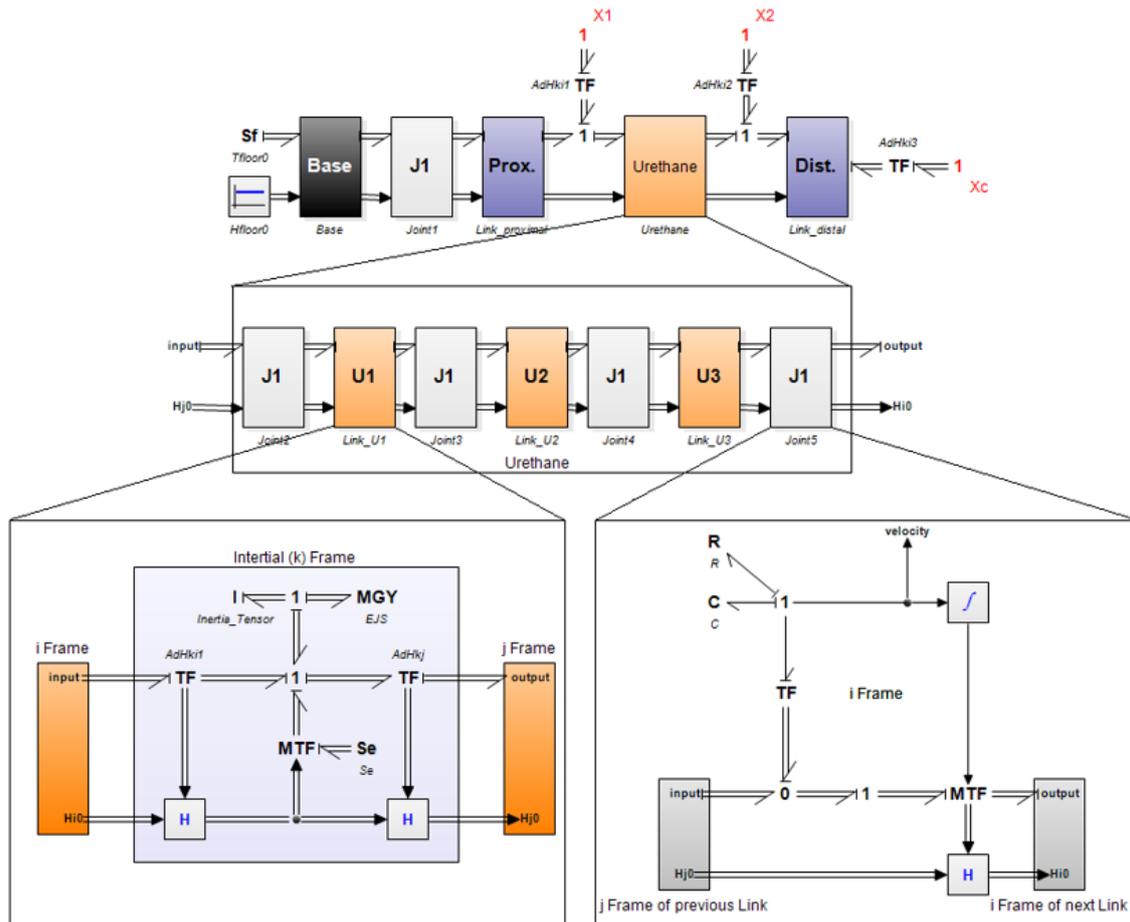


Figure 6.1: The full model as implemented in 20-sim, excluding all non-physical elements. The two core blocks are the translational (left) and rotational (right) coordinate changes, of which the translational block contains all inertial and gravitational information. These blocks were designed by D. Dresscher (Dresscher and Stramigioli (2010)) and were modified for this project. The Flexible Joint (Urethane block) is a series of these blocks in order to estimate the bending behavior of the joint.

Appendix B - Custom Blocks

StateCalc

The state calc block uses several geometrical identities to calculate the relevant angles required for the pseudo transformer. The block features 3 outputs: θ_1 , θ_2 and L_{12} , as shown in Figure 4.12.

```

parameters
  real global q_node;
variables
  real base_prox_angle;      //angle between the base and the proximal phalanx in the 0 frame.
  real prox_dist_angle;     //angle between the proximal and distal phalanges in the 0 frame.
  real base_dist_angle;     //angle the rope makes between points X1 and X2 in the 0 frame.
  real prox_dist_l;         //length L_12
  real reaction_angle;      //Angle deflection at point X1.
  real reaction_angle2;     //Angle difference between the string and distal phalanx.
equations
  base_prox_angle = input_q;
  prox_dist_angle = (atan2( (Hdist[3,4]-Hprox[3,4]) , (Hdist[1,4]-Hprox[1,4]) ));
  base_dist_angle = (atan2( (Hdist[3,4]-Hbase[3,4]) , (Hdist[1,4]-Hbase[1,4]) ));
  prox_dist_ropelength = sqrt((Hdist[3,4]-Hprox[3,4])^2 + (Hdist[1,4]-Hprox[1,4])^2 );
  reaction_angle = (pi + base_prox_angle - prox_dist_angle);
  reaction_angle2 = pi/2 + base_dist_angle - prox_dist_angle;

  Theta_1 = reaction_angle;
  Theta_2 = reaction_angle2 + (pi/2 - q_node);

```

Figure 6.2: The code for the StateCalc block. It uses the angular orientation of the various bodies with respect to the inertial frame to calculate the various angles required in the pseudo transformers, as well as the distance between X_1 and X_2

Pseudo-Transformer - X_1

The pseudo-transformer was made using custom elements and not implemented the same was as it is depicted in Figure 4.6. The main reason for this was that this is provided less clutter and allowed for some easy alterations. The transformers feature an if statement because the string can only exert a force in one direction.

```

parameters
  real global q_node;

variables
  real q_reaction;
  real q_force;

equations
  q_reaction = q_input/2;
  q_force = if q_input > pi then pi/2 else q_input/2 end;
  p2.e[1] = 0;
  p2.e[2] = 0;
  p2.e[3] = 0;
  p2.e[4] = if p1 > 0 then 2 * p1 * cos(q_force) * -cos(q_reaction) else 0 end;
  p2.e[5] = 0;
  p2.e[6] = if p1 > 0 then 2 * p1 * cos(q_force) * sin(q_reaction) else 0 end;

```

Figure 6.3: Code of the Pseudo Transformer at X_1 .

Pseudo-Transformer - X_2

```

parameters
  real global q_node;

variables
  real q_reaction;
  real q_force;

equations
  q_reaction = -(pi/2 - q_node) + q_input / 2;
  q_force = if q_input > pi then pi/2 else q_input/2 end;
  p2.e[1] = 0;
  p2.e[2] = 0;
  p2.e[3] = 0;
  p2.e[4] = if p1 > 0 then 2 * p1 * cos(q_force) * sin(q_reaction) else 0 end;
  p2.e[5] = 0;
  p2.e[6] = if p1 > 0 then -2 * p1 * cos(q_force) * cos(q_reaction) else 0 end;

```

Figure 6.4: Code of the Pseudo Transformer at X_2 . This block does one additional subtraction from the reaction force.

Pseudo-Transformer - X_c

```

parameters
  real global q_node; //angle between the rope and distal phalanx.

equations
  p2.e[1] = 0;
  p2.e[2] = 0;
  p2.e[3] = 0;
  p2.e[4] = if p1 > 0 then p1 * cos(q_node) else 0 end;
  p2.e[5] = 0;
  p2.e[6] = if p1 > 0 then p1 * sin(q_node) else 0 end;

```

Figure 6.5: Code of the Pseudo Transformer at X_c . This version does not have any variables in it, only a constant q_{node} .

Bibliography

- Colgate, J. and G. Schenkel (1997), Passivity of a class of sampled-data systems: Application to haptic interfaces, vol. 14, no.1, pp. 37–47.
- Controllab Products (2008), 20-sim.
<http://www.20-sim.com/>
- Dresscher, D., Y. B. P. B. J. B. and S. Stramigioli (2010), Modeling of the youBot in a serial link structure using twists and wrenches in a bond graph, *Proceedings of SIMPAR 2010 Workshops: International Conference on Simulation, Modeling, and Programming for Autonomous Robots. Darmstadt (Germany) November 15-16*, pp. 385–400.
- Franken, M., S. M. and S. Stramigioli (2012), Stability of position-based bilateral telemanipulation systems by damping injection, in *IEEE International Conference on Robotics and Automation*, pp. 4300–4306.
- Franken, M., S. S. S. M. C. S. and A. MacChelli (2011), Bilateral telemanipulation with time delays: A two-layer approach combining passivity and transparency, vol. 27, no.4, pp. 741–756.
- Hannaford, B. and J.-H. Ryu (2002), Time-domain passivity control of haptic interfaces, vol. 18, no.1, pp. 1–10.
- Hashtrudi-Zaad, K. and S. E. Salcudean (2001), Analysis of Control Architectures for Teleoperation Systems with Impedance/Admittance Master and Slave Manipulators, vol. 20, no.6, p. 419–445.
- Kapoor, S., P. A. V. K. M. J. M. T. (2014), Haptics Touchfeedback Technology Widening the Horizon of Medicine, vol. 8, no.3, p. 294–299.
- Lammers, B. (2017), *Design and realisation of a haptic interface between a reflex takkile and a omega 7 haptic device*, Master’s thesis, University of Twente | i-Botics.
- Lawrence, D. (1988), Impedance Control Stability Properties in Common Implementations, *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, pp. 1185–1190.
- Lee, D. and K. Huang (2010), Passive-set-position-modulation framework for interactive robotic systems, vol. 26, no.2, pp. 354–369.
- Orozco, Mauricio, J. S. A. E. S. E. P. (2012), The Role of Haptics in Games, *Haptics Rendering and Applications*, p. 217–234.
- Ott, C., R. M. and Y. Nakamura (2010), Unified Impedance and Admittance Control, *IEEE International Conference on Robotics and Automation*, pp. 554–561, ISSN 1050-4729.
- Robotics, R. (2014), RightHand Labs Robotic Grippers Tactile Sensors,
<http://www.labs.righthandrobotics.com/about-contact1>.
- Stramigioli, S., P. B. (1999), An interpretation of the Eulerian Junction Structure in 3D Rigid Bodies, *ResearchGate*.
- van Teeffelen, K. (2018), *Intuitive Impedance Modulation in Haptic Control using Electromyography*, Master’s thesis, University of Twente | i-Botics.
- Weir, D. and J. Colgate (2008), Stability of Haptic Displays, *Haptic Rendering*, p. 123–156.
- wiki, R. (2018), Wiki.
<http://wiki.ros.org/ROS/Introduction>