Development and evaluation of the Hybrid Hand

A prosthetic hand with force assist

L.R. Lommerse

Delft University of Technology

Development and evaluation of the Hybrid Hand

A prosthetic hand with force assist

by

L.R. Lommerse

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday January 28, 2021.

Student number:	4164415	
Project duration:	February, 2020 – January, 2021	
Thesis committee:	Prof. dr. ir. P. Breedveld,	Chairman
	Dr. ir. G. Smit,	Daily supervisor
	Dr. ir. E. van der Kruk,	External committee member

TU Delft

Abstract

Background Active hand prostheses aid people, who suffer from an upper limb deficiency, in performing activities of daily living. The two types of active hand prostheses available are the myo-electric prosthesis and body-powered prosthesis. Even though both allow a user to grasp objects, the downsides are major. The high actuation force required for the operation of body-powered prostheses and the high costs and low feedback of the myo-electric prostheses lead to high abandon rates. A hybrid solution could combine the best of both worlds: high output force and transparent feedback. In this study we introduce a body-powered prosthetic hand with force assist, that allows for user operation without muscle fatigue, while still providing the user with transparent feedback.

Method Design requirements are set up following parameters from literature. For this study the outer shell and closing mechanism of the 100-Dollar Hand have been used and adapted to purpose. An extensive mechanical analyses, a new conceptual design and the addition of a micro controller led to the new closing mechanism of the Hybrid Hand. The proposed prototype has been tested on a test bench and results have been compared to the conventional body-powered 100-Dollar Hand.

Results The Hybrid Hand decreases the actuation forces needed for a pinch force of 15 N with 33% in comparison to the 100-Dollar Hand. Moreover, the system increases the output pinch force at 80 N actuation force with almost 80%. With a weight of 320 gram and costs of under 200 dollar, the hand can amplify the pinch force with a maximum of 29 N. However, due to the low efficiency of the worm gear, the opening time of the hand (3.5 s) has to be improved.

Conclusion This paper presents the first 3D printed hand prosthesis with force assist that allows for user operation without muscle fatigue. With a low weight, low cost, high pinch force the prosthesis is accessible to and could aid people with an upper limb deficiency all over the world. Future research should further investigate the durability of the system and should introduce ways to improve and optimize the efficiency of the transmission of the system.

Contents

January 25, 2021

Development and evaluation of the Hybrid Hand, a prosthetic hand with force assist

Luuk Lommerse (4164415)

1. Introduction

1.1. Background

World wide people suffer from an upper limb deficiency, mostly caused by trauma or congenital defects. Different types of prostheses have been developed to aid these people in performing activities of daily living (ADL) [1]. These different types of prostheses that are available for upper limb extremities, can be divided in passive and active prostheses [2]. Passive prostheses overall need the other hand to make adjustments to the orientation and opening of the hand. Some passive prostheses only focus on aesthetic function, to make the hand look like a natural hand [3]. Active prostheses have a mechanism inside which converts an input by the user to the opening or closing of the hand. The main categories of active hand prostheses are body-powered and myo-electric hands, which both have their advantages and shortcomings.

Body-powered prostheses are powered with movement of another part of the body, mostly the shoulder (Figure 1 Top). A Bowden cable is attached to a shoulder harness so that the user can generate (input-) forces by moving their shoulder forward or extending their affected arm [4]. Because the cable is directly attached to the mechanism, the user has direct feedback in terms of input force and/or resistance. A downside of this technique is that is that body-powered prostheses mostly require high input force. These input forces range from 33 to 131 Newton (N) to generate a pinch force of 15 N [5]. The high actuation forces lead to user discomfort, exhaustion or even overuse [6], especially when constantly applying forces with the shoulder while holding an object. These factors contribute to a high abandon rate of prostheses [7]. Fatigue free cable operating force is reported to be at 20% of the maximum cable force of an user. For the average female this fatigue free operating force is reported to be 38 N. For the average male user this force is 66 N [8].

Myo-electric prostheses can overcome these problems by using an electro-motor to actuate the hand movement. As input, muscle activity of undamaged muscles is used (Figure 1 Bottom). Because the actuation is solely electric, feedback is fully dependent on visual feedback and the user has no clear indication of the input force [9, 10]. Actuation of a myo-electric prostheses mostly requires a lot of training and high concentration, because alternative muscles are used for the actuation. Moreover the myo-electric prostheses can be expensive ($6600 and up). The addition of an actuator and battery results in a weight which is perceived as 'high' by users [7, 11, 12].

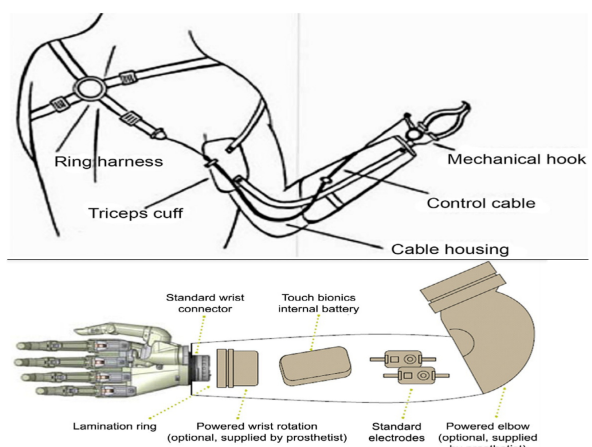


Figure 1: Types of active prostheses. Top: Working principle of the voluntary closing body-powered prosthesis. When the cable is pulled the hand closes. The user can pull the cable by extending his/her arm or by using their opposite shoulder [13]. Bottom: Example of a myo-electric prosthesis (i-Limb). The electrodes pick up a signal from the users muscles and translate this signal to opening and closing the hand[14].

1.2. Problem definition

Both body-powered prostheses and myo-electric prostheses have their downsides. The high actuation forces in body-powered prostheses lead to user discomfort, muscle fatigue and abandonment of the prostheses. Myo-electric prostheses are expensive, need lots of training and have no transparent feedback. In order to have the best of both worlds the two principles, high output force and transparent feedback, should be combined to form a hybrid solution.

1.3. Goal

The goal of this study is to design and validate a prosthetic hand with force assist, that allows for user operation without muscle fatigue, while still providing the user with transparent feedback (i.e. feedback forces on the cable).

1.4. Outline

This paper will start with the design requirements for the hybrid solution. After that, the mechanical analyses for the closing mechanism will be explained. Then, the design will be discussed and finally, the proposed prototype will be tested and evaluated.

2. Design requirements

2.1. General requirements

Before conceptual solutions have been made it is of importance to set design requirements. A summary for each aspect of the design the requirements can be found in Table 1. The requirements will be substantiated with literature in this Section. The principle of the Hybrid Hand prosthesis is to amplify the (input) forces of the user. For this to happen the force that the user puts in to the system needs to be measured and extra output has to be generated. This principle requires the use of an active prosthesis with basics of a voluntary closing body powered prosthesis with a Bowden cable and shoulder harness. For this study the design of the "100-Dollar Hand" (Figure 2) will be used and adjusted to fit the purpose [15].

Figure 2: The 100-Dollar Hand with Bowden cable. At the base of the hand a screw can be seen, so that the hand can easily be mounted to a socket.

2.2. Dimension requirements

The dimensions of the proposed solution should fit in a prosthetic hand. The outer shell of the hand is based on the design of the 100-Dollar Hand. The mechanical system should fit in the shell, preferably the control system as well. For the dimensions of the opening of the hand the 100-Dollar Hand had an opening of 60 mm. In this solution the hand opening should be at least 60 mm with an ideal goal of increasing the opening of the hand to 65 mm. To close a body-powered prosthesis, excursion of a Bowden cable is needed. This excursion is limited by the maximal body movement of an user. For current voluntary closing this excursion to close the hand is ranging up to 60 mm [16]. The maximum cable excursion using a harness is measured by Taylor to be 53±10 [17]. The design should not exceed this maximum range in terms of cable excursion.

Table 1: A summary of the design aspects with their requirements.

Requirements	Parameter	Required	Preverably
Dimensions	Integration in shell	Mechanical system	Mechanical system + control system
	Hand opening	> 60 mm	> 65 mm
	Cable excursion	< 53 mm	< 43 mm
Force	Pinch Force	>15 N	>25 N
	Transmission in comparison to 100-Dollar Hand	>90%	-
Control system	Power	6 Volt	-
	Lock	Locking capabilities	-
Weight	Weight hand	< 400 gram	< 300 gram
Costs	Material costs	< 200 dollar	-

2.3. Force requirements

The system should be able to amplify forces that the user puts in to the system. The goal of the design is for the user to use the hand prosthesis without muscle fatigue [8]. Most of the daily tasks can be completed with a pinch force of under 15 N, with some ranging up to 30 N [18, 19]. For the hybrid solution the maximal assisted output pinch force should be at least 15 N, preferably around 25 N. Because the closing of the hand is assisted by an electric solution, the transmission ratio of the actuation force to the output pinch force (without amplification) is not of great importance. Though to keep the complete system as efficient as possible, the transmission ratio (without amplification) should not be less than 90% of the 100-Dollar Hand transmission rate.

2.4. *Control system requirements*

Because the system needs to be wearable and be kept as light as possible we will use the smallest micro controller from Arduino [20]. All the sensor and actuator components should be compatible with this micro controller. To power all of the components and the micro controller a LiPo-battery pack will be used. With a higher voltage and power of this battery, size and weight will increase as well. The whole system should be able to be powered with a 6 Volt (V) battery, which converts to the actuator that has to be able to deliver the required forces at 6 V. To prevent the system from constantly drawing current from the power source when holding an item, the system should have some kind of lock. In terms of sensors the system should be able to measure the forces that the user puts on the Bowden cable.

2.5. *Weight*

Reasons for the rejection of electric powered prostheses and body powered prostheses include the excessive weight of these type of prostheses [7]. For the design of the Hybrid Hand the weight should be kept as low as possible. The 100-Dollar Hand has a mass of 250 gram, other body powered prostheses vary in weight from 220 gram up to 366 gram (without glove) [21]. The Hybrid Hand should remain in the range of weights of these body powered prostheses. Therefore the weight of the Hybrid Hand should be kept below the 400 gram, even aiming on 300 gram.

2.6. *Costs*

The 100-Dollar Hand is named after the fact that it can be produced for under 100 dollars. This makes the hand suitable for users in low resource countries. The proposed Hybrid Hand should still be available for people with a lower income. The materials needed to build a complete hand should not cost more than double the price of the 100-Dollar Hand. Other costs, such as equipment (tools and machinery) and labour will not be taken into account.

3. Mechanical analyses

3.1. *Closing mechanism*

Before the actual design process of the hybrid solution can be started, the body-powered version, the 100-Dollar Hand, has to be analysed. The mechanical solution of opening and closing the hand is a multiple-crank linkage system which is operated by a Bowden cable connected to the first crank. This crank is connected with a rotary point to the inner shell of the hand. Also attached to the Base crank are two 'floating' cranks (not connected to the shell of the hand). One for the finger part and one for the thumb part. Lastly two solid L-shaped cranks are situated in the fingers and the thumb of the hand. Both the finger and thumb crank have a rotation point on the shell of the hand. When the Bowden cable is operated (with a pulling motion) the base crank will rotate counter clockwise and a scissor-like motion occurs which allows the hand to be closed. To open the hand a spring is connected on the opposite side of the Bowden cable. This opposing force opens the hand when no user force is acting on the cable. On the right side of Figure 3 the cranks can be seen in the shell of the hand. This spring is an inefficiency of the system because the user has to overcome the counteracting spring forces before the hand can be closed.

3.2. *Model simulation*

To understand the underlying principle of the force transmission in the 100-Dollar Hand, Free Body Diagrams (FBD) have been made for each of the separate cranks of the system (Appendix A.). With all the directions of forces known, the closing

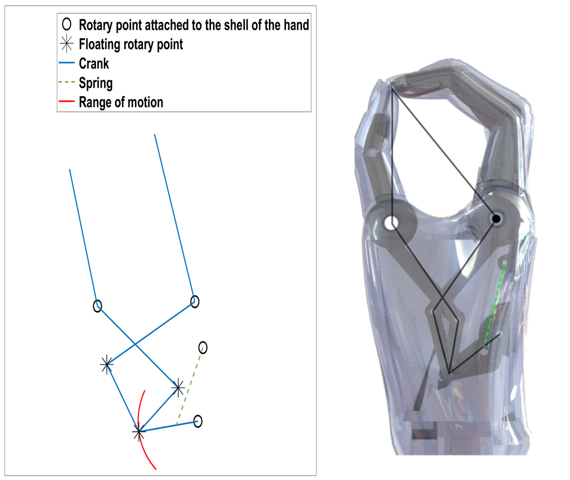


Figure 3: Analyses of the closing mechanism. Left: The mechanism of the simulation model plotted. Right: Solidworks version of the 100-Dollar Hand, lines have been added to clarify how the simulated version compares to the real model.

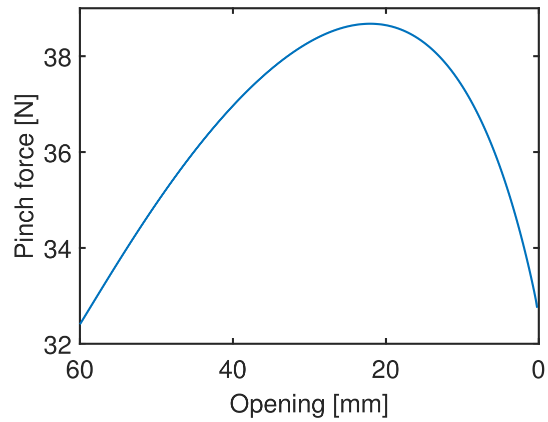


Figure 4: With an input force of 100 N the output pinch force has been plotted against the opening of the hand. At 60 mm opening the hand is completely open and at 0 mm the hand is completely closed.

mechanism of the hand has been modelled in 'MATLAB' (Figure 3 left). The output pinching force will be of importance when holding an item (in a static position), but not in the process of closing the hand. Therefore static analyses are chosen over dynamic analyses. In Figure 4 a 100 N input force has been used to plot the output pinch force against the opening of the hand (100-0%). The forces acting on the floating cranks separately, for an input force of 100 N, can be found in Appendix B.

3.3. Test set-up

For the validation of the model a pinch test has been carried out with the 100-Dollar Hand. In Figure 5 a schematic overview of the test set-up can be seen. The set-up consists of a spindle, that, when operated, changes the location of the cable (pull/-push). Attached to the spindle is a Linear Variable Differential Transformer (LVDT) which maps the horizontal location of the cable, and a load cell to sense the forces applied to the cable. To measure the pinch force, a sensor has been placed between the thumb and index finger. The thickness of the sensor is 10 mm, so the hand will apply forces in a 10 mm open position. Results can be viewed on an attached computer and are logged in a file (.txt). The actuation force, pinch force and horizontal location of the cable were used as outputs.

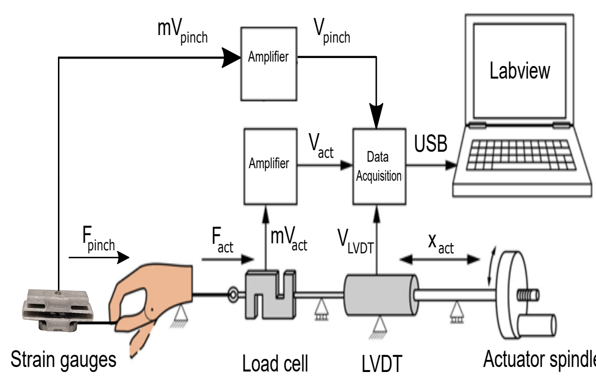


Figure 5: Test set-up used for testing the transmission ratio of the 100-Dollar Hand. Adapted from [21].

3.4. Validation

The parameters retrieved from the tensile pinch force test with the 100-Dollar Hand were then imported in the (simulation) model (Appendix C.). Two inputs were used for the model; the actuation force and the opening of the hand. The actuation force was used as an input to simulate the corresponding output pinch force of the hand. The 10 mm opening of the hand is of importance to determine the directions of forces of the mechanism internally. The force transfer in this position can be acquired from Figure 4. An input force of 100 N results in $\sim$ 37 N pinch force (Figure 6). Subsequently in this position the transmission ration of the input actuation force to the output pinch force can be determined to be approximately:

$$Transmission\ ratio = \frac{Pinch\ force}{Actuation\ force} \approx \frac{37}{100} = 0.37$$

With the actuation force and the opening of the hand as input for the model a comparison can be done with the real tensile pinch force test. In Figure 6 the comparison of the model with three tensile tests can be seen. The slope of all of the plotted lines are comparable, when fit. An estimated 15 N activation force is needed to actually close the hand, which is a result of the friction and the opposing spring in the system. The model itself does not go through the x-axis because the first actuation input force (from the tensile test) is higher than the opposing spring force. Resulting in the first data point being at approximately 2 N pinch force.

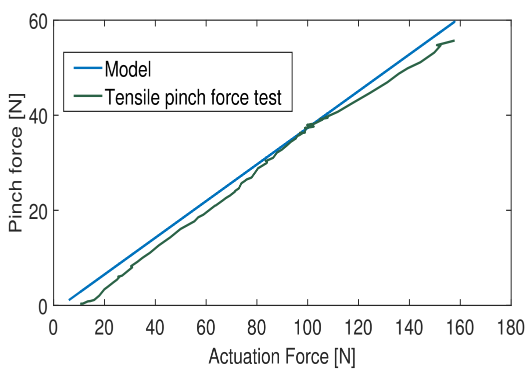


Figure 6: Tensile comparison of the model with the 100-Dollar Hand. The model slightly overestimates the pinch force for each actuation.

4. Design Hybrid Hand

4.1. Process

With the model as baseline and the design requirements as criteria, conceptual solutions can be generated. For the design process multiple methods from the Delft Design guide are used [22]. First, the problem has been divided in multiple sub-problems or components that the system should contain; the hybrid mechanism (method of adding forces), an outer shell, closing mechanism, actuator, transmission and locking mechanism. Solutions for each of these components have been summarized in a morphological overview (Appendix D.). The advantages and disadvantages will be discussed. For each component one solution will be chosen for the final prototype. A systematic overview of the final concept can be seen in Figure 8.

4.2. Outer shell

For the outer shell of the hand the design of a complete new shell and the use of an existing shell have been contemplated. Because the focus of the research should be on the amplification of the forces and the control of this amplification, an existing shell functioning as a basis has been chosen. The shell of the hand is based on the design of the 100-Dollar Hand which was originally designed for unilateral underarm amputees in developing countries. A test-case for this hand has been done in India, the size of the hand is based on an average Indian male [23]. Height of a person has a high correlation to the hand size of a person [24]. For India the average height of an adult male is approximately 165 cm and for a Dutch adult male approximately 183 cm [25]. Thus the average Dutch male is approximately 11 percent bigger than the average Indian male. For the scale factor of the hand this percentage has been rounded to 10 percent. Hence, to represent the hand-size of a Dutch male every component of the hand has been up-scaled with 10 percent in all directions. The shell was designed and adapted in 'Solidworks'. At the inside of the shell multiple structures have been added to accommodate sensors and mechanical components necessary for the hybrid design. The hand itself is composed of 2 shells which can be screwed together. The fingers and thumb are also linked with 2 separate shells.

4.3. Closing Mechanism

Before any changes have been made to the mechanical principle of closing the hand, the mechanism of the 100-Dollar Hand has been modelled MATLAB in Section 3. Due to the scaling and thus the increased dimensions of the shell of the hand the multiple bar linkage mechanism had to be changed as well to keep the ability to connect the cranks with each other. This was also needed to keep the 'hand-opening' above the required 60 mm and the efficiency of the input to output transmission approximately the same. When the shell was scaled by a 1.1 factor the multiple bar linkage mechanism had to be adjusted accordingly to keep the possibility to completely close and open the hand. The lengths of the cranks have been optimized using iteration. Both the opening of the hand and the transmission ratio had to satisfy the design requirements. All of the cranks of the mechanism have been laser-cut from 2 mm thick stainless steel. The parts that are attached to the shaft have a d-shaped profile to lock the rotation. The model of the 100-Dollar Hand has been adapted to represent the scaled version of the closing mechanism (Figure 7). A transmission ratio of 0.35 has been realised at 100 N of activation force. Which is within the required 90% from the 0.37 transmission ratio of the earlier model (Figure 6).

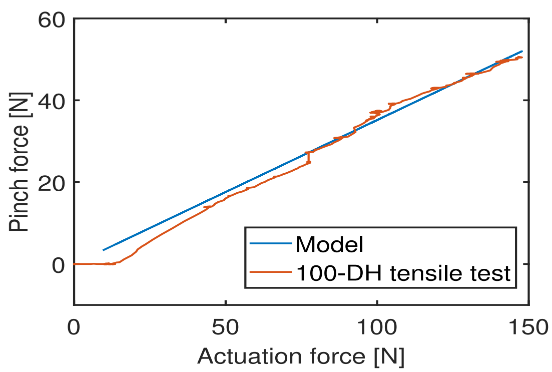


Figure 7: The scaled closing mechanism compared to the tensile test of the 100-Dollar Hand.

4.4. Hybrid mode

Due to the fact that the manual closing of the hand will be amplified by an electric actuator, two forces need to be summed to one output. Multiple solutions for this problem have been evaluated. To begin, the transmission of hybrid cars and bikes has been investigated. In these applications planetary gear sets are commonly used to add mechanical and electrical forces [26, 27]. The possibility of using planetary gears in the Hybrid Hand are limited, because of the small space available inside the hand. Furthermore, the speed of both inputs will change, depending on the gear ratios and input speeds. The variable speeds depending on the input velocities will reduce the transparency of the system. After that, series elastic actuators (SEAs) have been considered. These SEAs consist of an actuator component in series with an elastic element[28]. By using the actuator to control the different lengths of the spring the output force can be added to the input of the user. Due to the spring only being able to transfer pull forces (being extended), a secondary spring would be needed to allow the system to deliver forces both ways. To overcome this problem a mechanism has been designed with 2 parallel arms with a spring attached between both ends (Figure 8).

The first rotary arm is actuated by the user via a Bowden cable and the second arm by an actuator. By changing the orientation of the second rotary arm (motor side) in comparison to the first one, the amplified output force can be controlled. With the traditional closing mechanism of the 100-Dollar Hand, a spring was needed to open the hand when not applying any force. Before being able to close the hand the spring force has to be overcome. With the hybrid mode this opening-spring is not needed, thus in theory decreasing the actuation force.

4.5. Actuator design

For the application a small, lightweight and powerful motor is needed. Multiple options were possible for the use of an electric actuator. To choose between a rotational and linear actuator, the pros and cons of both systems have been weighed. Linear actuators are mostly less powerful or more bulky than the rotation option. To create a linear movement a rotational actuator is often combined with gears to translate the movement. The linear actuators that are powerful enough to generate a pinch force of 20 N, do not fit in the outer shell of the scaled 100-Dollar Hand. Therefore the remaining choices are the rotational actuators, which come in 3 versions: DC motors, Servo motors and Stepper motors. Stepper motors have high torque at low rotational speeds, which is a desirable property for a prosthetic hand. However, Stepper motors are more expensive than the other two options and very limited in terms of miniature dimensions. Moreover, a Stepper motor is too heavy for this application (60 gram and up) [29]. DC motors are continuous rotation motors which can be controlled with a PWM signal (Pulse Width Modulation). The speed

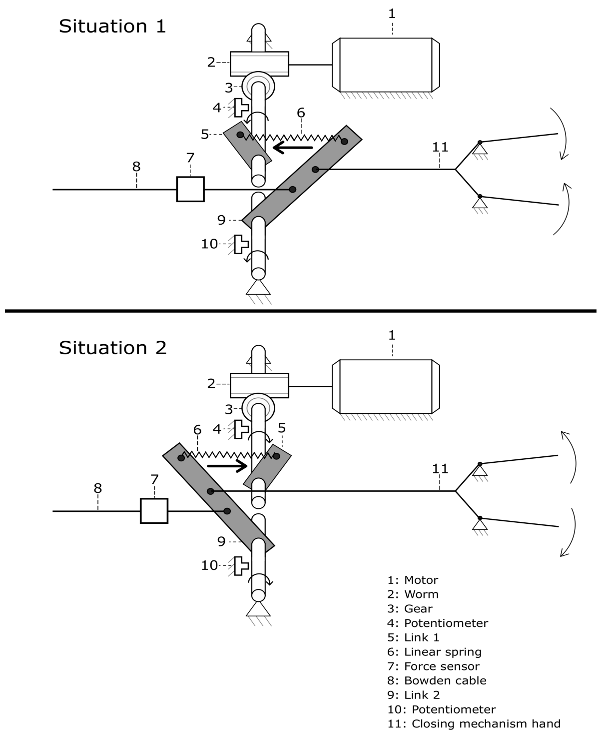


Figure 8: A systematic overview of the final solution. In Situation 1 the motor helps the user to close the hand and amplify the actuation forces of the user; the spring-force on the base crank of the user is directed to the left. In Situation 2 the spring force on the base crank of the user is directed to the right, thus closing the hand.

of the DC motor can be controlled by varying the pulse width. DC Motors have a high starting torque and are quick in changing direction [30]. Both are desired properties for our system, making the DC motor a viable option. The last option considered, is a Servo motor, which can be seen as an assembly of 4 components: DC motor, gears and a control-system. Servo motors can be accurate position motors, but are limited in range and have no feedback system when slipping due to high torques.

For our application a micro metal geared DC motor has been used from POLOLU [29]. With a cross section of 10×12 mm, a length of 26 mm and a relatively high stalling torque at 6 V (up to 12 kg*cm), these motors were suitable for our application. The standard motor clamp available from the manufacturer was not strong enough to keep the motor in place under the radial and axial forces. For the DC motor a custom motor-bracket has been designed (Figure 9), the bracket allows the motor to be clamped to the wall using nuts and screws. On both sides of the motor nuts can be placed in the bracket. At the inner side of the bracket a solid block has been added, which prevents the motor from moving under the potential axial forces.

After an analysis the motor (Appendix E.) an axial bearing has been added to the motor shaft. The bearing refrains the motor from being stalled due to too much internal friction from axial forces, which come to play due to the worm gear. Because the axial bearing is enclosed by the motor bracket it also keeps the motor shaft from bending and stalling due to radial forces. When holding an item, a constant force amplification is needed and the actuator would have to maintain its position meanwhile constantly drawing power. To prevent this some type of lock should be added.

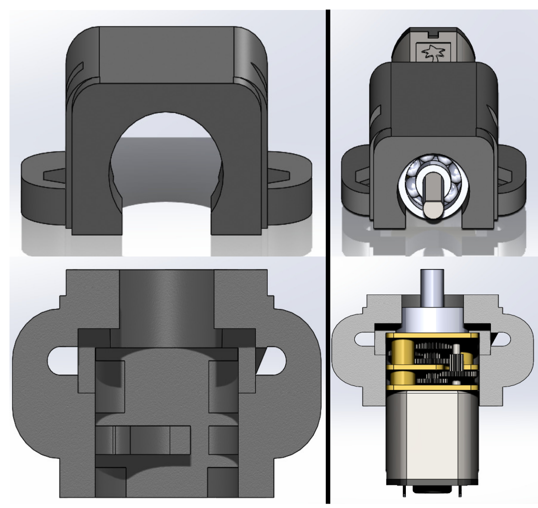


Figure 9: Left: Designed motor bracket with inner blocks Right: Assembly of the motor bracket and the motor. The round shape at the front of the bracket allows the axial bearing to also refrain the motor shaft from bending due to radial forces.

4.6. Transmission design

Due to all of the components, a motor will not fit horizontally in the hand, therefore a 90 degrees translation of the rotational movement is necessary. Several gear types are available to realise a 90 degrees translation. These options have been summarized in Appendix D. Although worm gears commonly have a lower efficiency than helical gears or

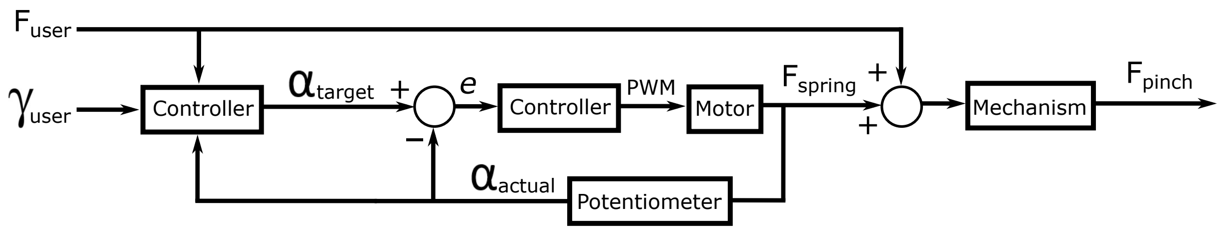


Figure 10: Control block scheme. The controller calculates the target location for the motor from the angle of the user base crank (γ_{user}), the user actuation force (F_{user}) and the actual position of the motor base crank (α_{actual}). The motor control loop is acting as a servo motor control loop. The output pinch force is a result of two forces acting on the user base crank: user actuation force and spring force.

bevel gears, the self-locking ability is a big advantage in the Hybrid Hand [30]. By having a lock, the motor can be turned off when the correct force or position has been reached, instead of constantly stalling and drawing current. With the self-locking capabilities of the worm gear the need for a custom lock design is unnecessary. This would make a worm gear the best option for the translation of actuator motion. The second component transmitting the torque to the user side of the mechanism is a spring. For this application torsion springs and tension springs can be used. Tension springs have a wider variety available. In addition a solid attachment of a tension spring is easier to realize and by varying the attachment point small tweaks can be done to accommodate a better force transfer.

4.7. Sensor design

For the system to accurately control the amplified pinch force, multiple sensors are needed. To determine the actuation force the user puts in to the system, a FUTEK miniature load cell has been connected the Bowden cable [31]. Two variables that dictate the extra force the amplification system delivers, are the length and orientation of the spring. With these variables known the amplified output pinch force can be determined. For the orientation and length of the spring the orientation of both the base crank of the user and the base crank of the motor are needed. Potentiometers are a cheap and reliable way of measuring the angle of the shaft and thus, the cranks. For each of the shafts a potentiometer has been adapted to fit the limited space. The control element (shaft) of the potentiometer has been trimmed. A small cap with a D-shaped hole in it has been designed to slide the potentiometer on the shaft (Appendix F.). The combination of the actuator with a potentiometer on the shaft after the worm gear transmission makes it possible to control the position of the crank and spring on the motor side. The control block scheme for the system can be seen in Figure 10. Here F_{user} will be measured with the load cell, γ_{user} with a potentiometer.

5. Parameter specification

5.1. Model baseline

To determine suitable dimensions and parameters for the system, a model of the complete system has been made. With this model the output pinch force can be predicted for each position of both cranks. The variables that are used as input are the length of the cranks (motor side and user side), the spring characteristics and the efficiency of the worm gear. As a baseline to create a moment arm as large as possible, the attachment point of the spring on the user side crank has been maximized to still fit into the hand. This resulted in the attachment point 35 mm from the rotation shaft. For each of the positions of the hand, the transfer function from user input moment to output pinch force has been retrieved from the model created during the analyses of the closing mechanism of the 100-Dollar Hand (Appendix G.). Each of the parameters optimized will be mentioned in this section.

5.2. Spring

From the baseline, the second step was to find commercial available springs which were able to fit between both shafts. The spring has to be capable of both storing enough energy to give the required amplified pinch force of at least 15 N and have a safety factor to keep the spring from being plastically deformed. The last requirement was to have a spring constant, C [N/mm], that was high enough to store the force in the limited space available. The catalogue of springs of TEVEMA Technical Springs has been used to find available springs [32]. Three input filters were used to find possible springs:

- Resting length, $L_0 < 25$ mm
- Maximal force, $F > 100$ N
- Maximal spring length, $S_n > 6$ mm

The spring chosen was the T41920 which has a L_0 of 22.1 mm, a C of 20.4 N/mm, a S_n of 6.8 mm and an initial tension of 20.8 N [32].

5.3. Length motor base crank

For the length of the motor base crank the L_0 is of importance. The cranks must be able to pass each other without the spring being pushed of the attachment point. However, the attachment points should not be too far from each other, which leads to a high pre-tension of the spring. With the attachment point of the spring on the user base crank at 35 mm and an M3 screw as attachment axis, the length of the spring attachment is at $35 + \frac{1}{2} * 3 = 36.5\,mm$. The L_0 of the spring is 22.1 mm, thus for a pre-tension that is as small as possible the attachment point of the motor base crank will be at $36.5 - 22.1 + \frac{1}{2} * 3 = 15.9\,mm$. To prevent the spring from being pushed of the attachment axis, this has been set to 14 mm.

5.4. Motor and transmission specifications

If all of the spring parameters are known, the output pinch force can be plotted for each of the motor positions. For this simulation the closed hand position has been used, because maximal possible distance between the two attachment points (length spring) is the smallest. The motor (gear) specification has to be done together with the worm gear, the combination of these two define the transmission ratio. For the worm gear a ratio of 1:20 will be used, for lower ratios the locking capabilities of the worm gear are limited. With a higher ratio comes a larger radius of the gear, which will not fit in the hand. From Mädler a worm and gear has been ordered with a transmission ratio of 20 [33]. Worm gears have an efficiency up to 95% [34]. To get an understanding of the output forces for different motor torques, the system has been simulated. For the closed position of the hand and a worm gear efficiency of 0.90 the amplified output forces have been plotted (Figure 11). On the X-axis of this figure we

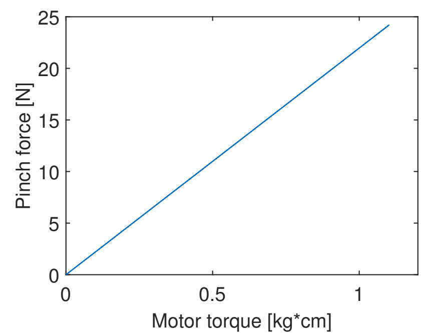


Figure 11: Motor torque plotted against the pinch force amplification, with a worm gear efficiency 0.9.

have the motor torque in kg*cm. On the Y-axis we have the output pinch force in Newtons. The line does not go to infinity because of the limitations in elastic spring deformation and the limited maximal length between the two attachment points.

To reach a pinch force of 15 N a motor torque of 0.69 kg*cm will be needed. To allow for extra inefficiencies from POLOLU a micro DC gear motor with 0.86 kg*cm and 590 RPM (rounds per minute), the 50:1 Micro Metal Gearmotor HP. To close the hand the motor base crank has to travel 85 Degrees. The motor specifications state 590 RPM and with a worm gear transmission of 20 this is approximately 0.5 rounds per second. To close the hand the motor has to travel approximately a quarter round, which is equal to 0.5 seconds.

6. Prototype development

6.1. 3D printed parts

Prototyping of the outer shell and fingers has been done by additive manufacturing in the form of 3D printing. For this process an 'Ultimaker 3 Extended' printer has been used [35]. In terms of the main printing material for the process ABS (Acrylonitril-butadieen-stryeen) and

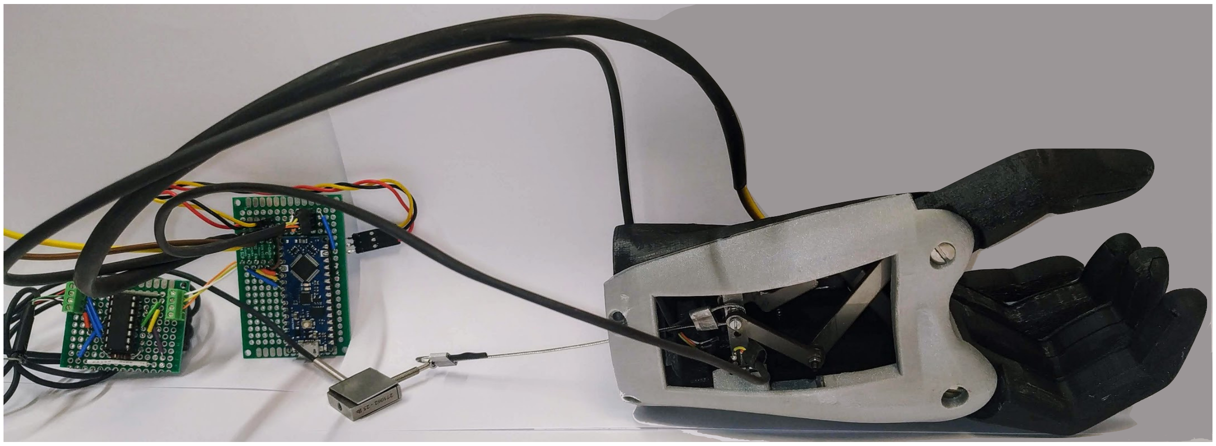


Figure 12: Complete system of the Hybrid Hand. Via a viewing hole in the outer shell the mechanism can be observed. Three cables come from the hand to the soldered PCBs, 1 for the motor and 2 for the potentiometers.

PLA(Polylactic acid) have been considered. For the printing process of the 100-Dollar Hand, PLA had better and more consistent 3D printing results, making it the better option for this prototype. In terms of support material Ultimaker's Breakaway material and PVA (Polylactic acid) have been looked into. Where breakaway can be removed using pliers, PVA dissolves when it comes to contact with water. The shell has some small overhanging structures allowing all mechanical components to be attached. To prevent these structures from being damaged by pliers the PVA support material has been chosen. The water soluble characteristics make it straight forward to remove support material. Before the complete prototype has been printed, a test model containing only the part with the added internal structures has been printed. From the test model the measures and precision could be observed. Because all the mechanical components inside the shell have to be attached in a small space with enclosed walls, a step-by-step assembly protocol has been made. Each of the components attachment places, has been analysed and optimized so that all of the parts can be disassembled as well. In Figure 12 the complete prototype of the Hybrid Hand can be seen.

6.2. Conventional fabricated parts

For some of the parts no 3D printed alternative was feasible, due to the limitations of yield strength and tensile stress. These parts were ordered or manufactured with conventional methods. The cranks of the closing mechanism of the hand have been cut using a laser cutter. For all of the cranks a 2 mm thick stainless steel sheet has been used. For the attachment of these cranks M2 screws are used. To keep the attachments of the base cranks with the spring from bending, M3 screws have been used on these points. At 1 side of each of the shafts, the user-shaft and the gear-shaft, a D-shaped profile has been made using metal milling. The base cranks of the mechanism are connected to the shaft with this D-shaped profile. The worm and the worm gear are attached to the shafts with set screws. One for the worm on the D-shaped motor shaft and 2 for the round gear-shaft. To increase the resistance of the attachment on the shaft spots of the set screws small imprints have been made on the shaft with a centre bore.

6.3. Actuator tests

When the prototype was assembled, motor tests have been done. The 50:1 Micro Metal Gearmotor could only deliver a pinch force of 3 N. On the site of Mädler an efficiency of the worm gear of 0.53 was stated. After a new simulation with an efficiency of 0.53 a motor torque of 1.9 kg*cm was required to get an amplified output pinch force of 23 N. The first micro DC gear motor from POLOLU capable of reaching a stalling torque of at least 1.9 kg*cm is the 210:1 HP micro metal gear motor. This motor has been tested with a load before and after the worm gear transmission (Appendix E.). From this tests resulted that the actual stalling torque was 1.5 kg*cm and the worm gear efficiency was actually around 15%. After optimizations of the design, an efficiency of around 25% was realized. To reach the required pinch force of 15 N two options were considered: get a stronger motor with a higher stalling torque or decrease the radius of attachment of the closing mechanism on the user base crank. When decreasing the radius the of the closing mechanism the maximum opening of the hand decreases as well, which could limit the user in performing daily tasks. For the prototype a motor with a higher gear transmission has been chosen. A 380:1 HP micro metal gear motor with a stated stalling torque of 5.5 kg*cm has been used for the prototype.

6.4. Electronics and sensor calibration

To control the system, a micro controller in the form of an Arduino Nano Every has been used. This model is the smallest available from Arduino, with dimensions of 45x18 mm. It can be powered with 5 V and has a clock speed of 20 MHz [20]. The Arduino Nano Every has enough ports for our sensors, three digital ports and three analogue ports are needed. To measure the input cable force of the user a Miniatur S-Beam Jr. Load Cell from FUTEK has been used [31]. For the Arduino to be able to read the sensor values an amplifier has to be made. For this an INA125p amplifier has been used. The electric schemes, and the PCBs soldered can be found in Appendix H. To control the DC motor the motor driver (DRV8838) from POLOLU has been used. With this motor driver the speed and direction of the DC motor can be controlled. The motor driver has been soldered to a second PCB, on this PCB the Arduino has been soldered as well. Each of the sensors can be plugged in to the PCB, so that the exchange of a sensor can be done relatively easy (Figure 13).

Each of the sensors have been calibrated. The potentiometer connected to the motor shaft has been calibrated using photo analyses. For 5 positions of the sensor a photo has been made and the angle of each of these positions has been derived using Inkscape. Because the potentiometers are linear, a function fit could be done using MATLAB. For the calibration of the load cell weights were used. One side of the load cell was clamped and on the other side the attached weights were increased. From these results a fit again could be applied using MATLAB.

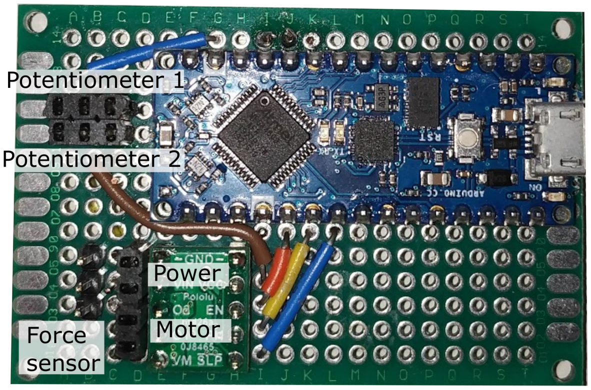


Figure 13: PCB with the motor driver and the Arduino. For each of the sensors a connection plug has been made. This way broken sensors can be replaced quickly.

6.5. Control program

The program to control the system has been written in the Arduino Integrated Development Environment (Arduino IDE). Most of the functions used in this software are based on the C and C++ programming language. To start, a function for the position control loop of the motor has been made. This function makes it possible to control the position of the motor base crank (with a stated allowable error). This is somewhat like a Servo motor, but in this system the position controlled is after the worm gear transmission. After having a position control for the motor it is of importance to know what location the motor needs to go to. To control the output pinch force the user actuation force is leading for the behaviour of the system. For the system 3 behavioural states of the user have been defined:

1. The user wants to open the hand or keep it open
2. The user wants to grab something, wants to close the hand
3. The user wants to apply force, hold an item

For each of these situations a threshold in terms of force applied by the user has been defined. When a user wants to open the hand he or she would remove the forces on the cable. While performing tasks of daily living, small forces can be present.

Therefore the threshold for the user actuation force of opening the hand/keeping the hand open can not be too low. For this first situation the actuation force of the user should be below 3 N. If this is the case the motor will set the motor crank to 45 degrees, which converts to the open position of the hand. For the second situation where the user wants to close the hand without actually amplifying pinch force, test results of the 100-Dollar Hand have been used. To close the 100-Dollar Hand an activation force of approximately 15 N was needed (Figure 6). The hybrid system should require less force to close the hand, while still remaining the closing of the hand as a viable state. While the user cable force is between the 3 N and 10 N and the hand is not closed, the motor will move the motor crank to the position of the user plus 5 degrees. This way the motor will be passively closing the hand without applying to much force when an item is grasped. The 5 degrees angle would in theory amplify the pinch force of the user with 1 N.

In the last situation if the user puts more force on the cable than 10 N the amplification state of the system will be activated. In this state from the position of the user base crank the transfer ratio activation force to pinch force will then be derived. This will be done by converting the angle to an index of an array. In this array all of the transfer ratios (input to output) are stated. The array has a resolution of 1 degree. The expected pinch force of the user will then be calculated by multiplying the activation force with the transfer ratio.

$$F_{pinch_{user}}[N] = F_{act}[N] * H_{ratio}$$

The system then multiplies this pinch force with an amplification factor to find the amplification pinch force.

$$F_{pinch_{motor}}[N] = F_{pinch_{user}}[N] * (f_{amp} - 1)$$

To prevent the system from having an overshoot of the desired pinch force the user wants, a kp value has been implemented for the $F_{pinch_{motor}}$. This kp is a value between 0 and 1 and forces the system to gradually deliver the desired pinch force amplification. The lower the kp the more steps the system takes to reach the eventual desired amplification pinch force. The formula for the amplification pinch force then becomes:

$$F_{pinch_{motor}}[N] = kp * F_{pinch_{user}}[N] * (f_{amp} - 1)$$

The corresponding location of the motor will be derived from a quadratic fit (Figure 14). An accuracy of around 2 N can be seen for most of the base crank positions. Because the data has been modelled and down-sampled with a resolution of 1 degree this error is acceptable. For each position of the user the parameters of this polynomial fit are slightly different. Each of these parameters have been stored in an array and will be derived when needed. The target position of the motor crank has been calculated with:

$$\theta_{target}[Deg] = A * x^2 + B * x + C$$

Here A, B and C are the parameters derived from the function fit. X is the desired motor pinch force amplification. To keep the motor from stalling, a maximalamplification pinch force can be put in to the system.

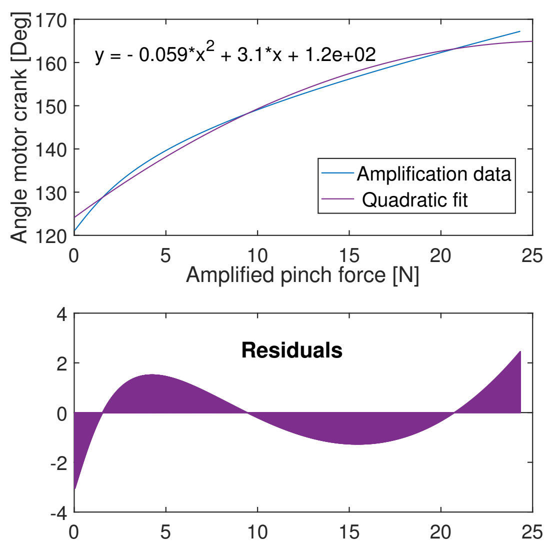


Figure 14: Example of a quadratic fit for the angle of the user crank where the hand opening is 10 mm. The residuals of the fit can be seen in the second plot. An accuracy of around 2 N can be seen.

7. Evaluation tests

7.1. Approach

To ensure the control system would behave as expected, static simulator tests have been performed.

The prototype experiments have been carried out to evaluate the performance and check whether the design requirements have been met. First of all, the weight, opening width and maximal pinch force amplification have been analysed. After that, three types of experiments have been executed to test the amplification performance: the work required to close the hand and achieve a pinch force of 15 N, an amplification pinch force test and a quick release test. The test set-up used for the performance tests is the same as for the test of the 100-Dollar Hand and can be seen in Figure 5. Before testing the sensors have been calibrated using weights. All of the tests have been performed three times and for each of the tests a test protocol was followed to ensure similar circumstances.

7.2. Virtual tests

To ensure the system calculated the correct locations for the desired amplification, some static simulator tests have been performed. With simulator software of Tinkercad, 4 test cases have been evaluated [36]. First, with a 0 N force input the motor should open the hand. Then, with a 5 N force input the motor should passively close the hand. After that, with an input force of 15 N the motor should move to a set target location to amplify the pinch force. Finally, a test has been done with a set maximal pinch force amplification of 15 N. The input force of 34 N would lead to an amplification pinch force of 21 N without the set limit. After all the virtual tests were successful, the actual prototype of the hand has been tested. The logs of the virtual tests can be found in Appendix I.

7.3. Mechanical properties

The maximum opening of the hand is 62 mm. In terms of weight, the complete prototype excluding battery weighs 320 gram. Without the control system (Arduino and PCBs), this comes to 270 gram. To confirm that the hybrid system is able to amplify the pinch force sufficiently, a test has been carried out measuring the maximum output pinch force. With an increasing position of the motor crank the output pinch force has been reported until the motor stalls (Figure 15). The maximal pinch force the motor can deliver is 29 N. To have an idea of the energy that the user puts into the system to close the hand and deliver a pinch force of 15 N, the work can be determined. The amount of work it takes to close the hand can be found by integrating the required activation force over the cable excursion. For closing the hand 156 Nmm is required. For closing the hand plus a pinch force of 15 N, 202 Nmm is required.

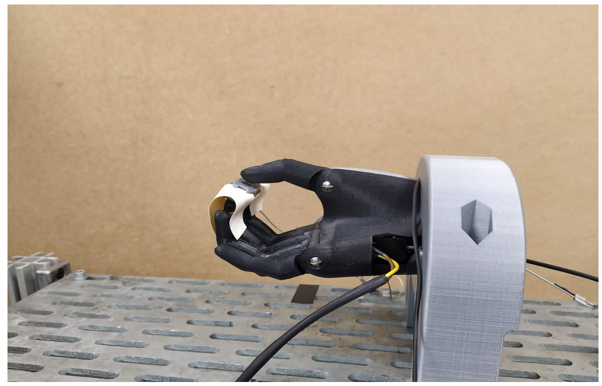


Figure 15: The prototype clamped to the test set-up, by increasing the angle of the motor crank the maximal pinch force can be tested.

7.4. Amplification test

To test the actual pinch force amplification the results of the 100-Dollar Hand pinch force test has been compared to the hybrid prototype. The parameters of the control system have been adapted to test the most viable parameter set-up. The standard parameters used in the test are: $f_{amp} = 1.5$, $kp = 0.9$, and an allowable error of the motor position of 2 degrees.

The maximal amplification for this standard test has been set to 15 N pinch force. For each of the tests one of these parameters has been changed. First a baseline test has been done with the standard parameters, results can be seen in Figure 16. To check if the accuracy of the motor position can be increased, a maximal allowable error of the angle of the motor crank of 1 degree has been used in Figure 17. All the values have been set to the standard and the kp factor has been decreased to 0.5 in Figure 18. This should allow the system to go to the required pinch force in more steps, thus slightly slower. For the next test (Figure 19) only the amplification factor has been increased from 1.5 (thus increasing the pinch force with a factor 0.5) to 2. For the last amplification pinch force test all of the optimal parameters have been combined.

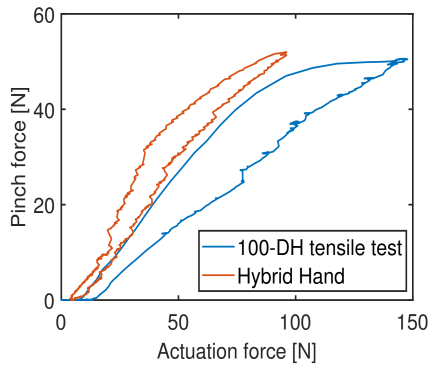


Figure 16: Amplification pinch force test for the parameters: $f_{amp} = 1.5$, $kp = 0.9$, and an allowable error of the motor position of 2 degrees.

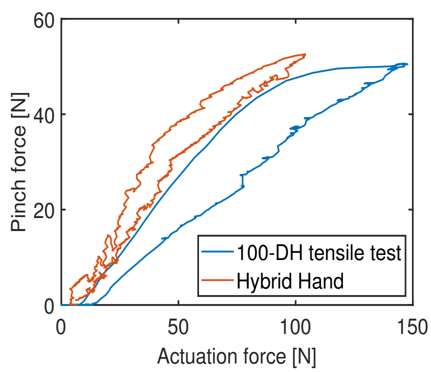


Figure 17: Amplification pinch force test for the parameters: $f_{amp} = 1.5$, $kp = 0.9$, and an allowable error of the motor position of 1 degree.

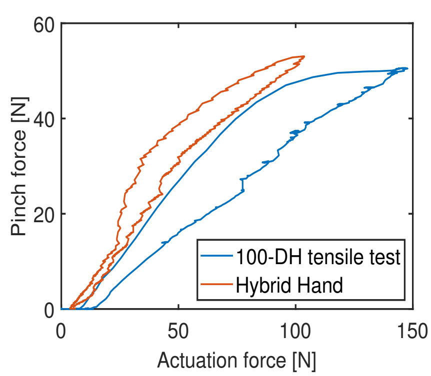


Figure 18: Amplification pinch force test for the parameters: $f_{amp} = 1.5$, $kp = 0.5$, and an allowable error of the motor position of 2 degrees.

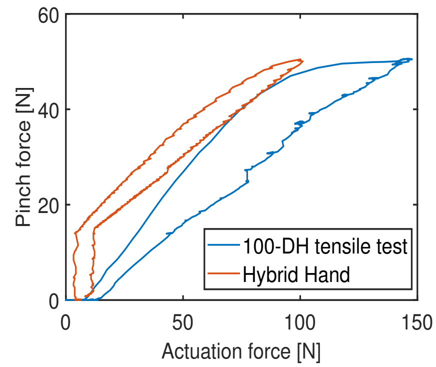


Figure 19: Amplification pinch force test for the parameters: $f_{amp} = 1$, $kp = 0.9$, and an allowable error of the motor position of 2 degrees.

$f_{amp} = 1.5$, $kp = 0.5$, and an allowable error of the motor position of 2 degrees. The limit of maximal amplification of pinch force has been set to 20 N (Figure 20).

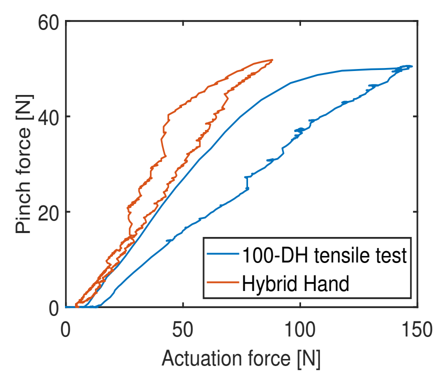


Figure 20: Amplification pinch force test for the parameters: $f_{amp} = 1.5$, $kp = 0.5$, and an allowable error of the motor position of 2 degrees. The limit of the maximal amplification of pinch force has been set to 20 N.

7.5. Quick release test

A quick release test has been done to test the delay of the motor amplification when the user suddenly removes all the actuation force. To simulate this behaviour, a pinch force of 30 N has been applied to the sensor. The actuation cable has been cut using pliers (Figure 21). The delay of the system can be seen by comparing the actuation force (Approximately 0 N after cutting the cable) and the pinch force. Results of this test can be seen in Figure 23.

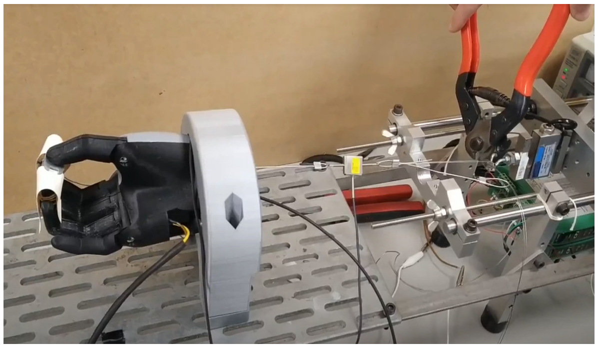


Figure 21: Test set-up of the quick release test. With a pinch force of 30 N, the cable is cut using pliers.

8. Discussion

8.1. Approach

The test results will be evaluated and compared to the design requirements and to the body powered 100-Dollar Hand. Even though the 100-Dollar Hand has been scaled up by 10 percent, this comparison is still viable, due to the new closing mechanism (without actuator) being approximately as efficient as the 100-Dollar Hand. The Arduino script used for the tests can be found in Appendix J.

8.2. Amplification factor

First of all, all of the amplification tests have been plotted in one figure to compare them to each other (Figure 22). Here the 100-Dollar Hand tensile pinch force test is plotted in dark blue and the Hybrid Hand with the standard parameters for the control system is plotted in brown. Because the hybrid system should amplify the pinch force the slope of the Hybrid Hand is expected to be steeper than the 100-Dollar Hand. With the 1.5 amplification factor, the slope is expected to be approximately 1.5 times as high as the slope of the 100-Dollar Hand. Following from a linear fit, the slope of the 100-Dollar Hand data is 0.38 where the slope of the standard Hybrid Hand parameters is 0.63 and thus translates to a 1.65 amplification factor in comparison to the 100-Dollar Hand. The slightly higher amplification factor can be explained by the model overestimating the actual pinch force generated by the user and inaccuracies of the control system. The control system uses a quadratic fit which has some inaccuracies (Figure 14). Moreover the accuracy of the motor crank is 2 degrees both ways. Following this theory, we expect that the amplification factor of the test with a motor accuracy of 1 degree in comparison to the 100-Dollar Hand will be closer to the expected 1.5 factor. The slope of the test run 'Hybrid Hand error=1 degree' is 0.57 which converts to an amplification of exactly 1.50. Though when we look at the plot of this test run solely, we can conclude that by making the motor crank angle 'more accurate' the system is more unstable due to more overshoots (Figure 17). Especially in the lower pinch force region, where the motor has less resistance of the spring (spring is less tensed), the motor system has some overshoot.

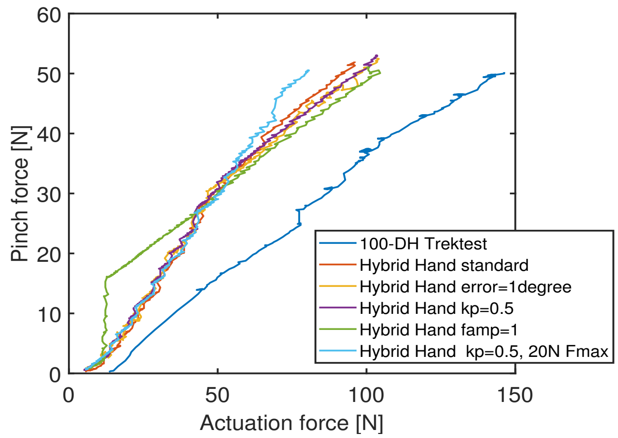


Figure 22: All of the pinch force amplification tests plotted in one figure. For readability only the closing of the hand has been plotted. The complete cycles can found in Section 7. Evaluation tests.

8.3. Maximal amplification

The required maximal pinch force amplification of the system was 20 N. If we look in Figure 22 at the highest pinch force of the last evaluation test ('Hybrid Hand kp=0.5, 20 N max amp'), a pinch force of 50 N at 80 N actuation force has been reached. When compared to the pinch force of 28 N of the 100-Dollar Hand at an actuation force of 80 N, a amplification of 22 N has been reached. For this test the amplified pinch force was limited to 20 N. For a test with no limit on the motor force (and an amplification of 1.5) a pinch force of 55 N was realised at 80 N actuation. This translates to a 27 N amplification.

8.4. Quick release

Results of the quick release test can be seen in Figure 23. From this figure we can derive the delay of the system, in other words: the time it takes for the system to react to the sudden drop in actuation force. It takes approximately 300 milliseconds for the system to notice the drop in actuation force (horizontal line). Before all the pinch force applied is removed from the fingertips, it takes another 1.5 seconds. The total time from cutting the cable until all the pinch force is removed is then 1.8 seconds. For a pinch force lower than 30 N this is expected to be quicker, due to a lower motor amplification and thus a smaller motor crank angle to travel. For safety measures this response time could be too high. However, because the motor crank is connected to the user base crank, the hand can always be forced open by pushing the fingers and thumb away from each other. The spring tension has to be overcome to realise this.

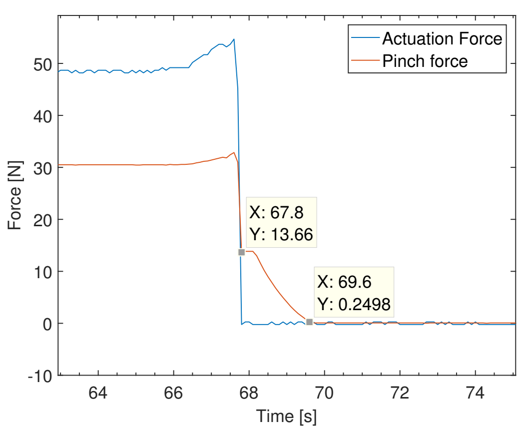


Figure 23: Results of the quick release test. With start conditions: motor amplification of 14 N and a total pinch force of 30 N.

8.5. Summary and assessment design requirements

The mechanical properties and dimensions can be summarized in a table in comparison to the 100-Dollar Hand (Table 2). For the values in this table, the last pinch force amplification test, with $f_{amp} = 1.5$, $kp = 0.5$, and an allowable error of the motor position of 2 degrees is used. The limit of maximal amplification of pinch force has been set to 20 N. The opening of the Hybrid Hand is slightly bigger than the 100-Dollar Hand, meeting the required hand opening of 60 mm. For a pinch force of 15 N an actuation force of 30 N is required for the Hybrid Hand. With a 80 N actuation force a pinch force of 50 N is reached. At 80 N this pinch force is amplified with 22 N in comparison to the 100-Dollar Hand. The proposed prototype meets the design requirements in terms of hand opening (62 mm>60 mm) and maximal pinch force amplification (27 N>15 N). However in terms of dimensions requirements, the control system does not fit in the hand, due to the dimension of the PCBs and Arduino. The assessment of all the design requirements is reported in Table 3.

Table 2: A summary for the characteristics of the Hybrid Hand and the 100-Dollar Hand [23].

	Hybrid Hand	100-Dollar Hand
Mass (g)	320	250
Opening width (mm)	60	60
Actuation force at 15 N pinch force (N)	30	47
Pinch force at 80 N actuation force (N)	50	28
Closing of the hand (Nmm)	160	280
Closing of the hand + 15 N pinchforce (Nmm)	200	290

Table 3: Assesment of the requirements.

Requirements	Parameter	Required	Preverably	Result
Dimensions	Integration in shell	Mechanical system	Mechanical system + control system	Mechanical system
	Hand opening	> 60 mm	> 65 mm	62 mm
	Cable excursion	< 53 mm	< 43 mm	30 mm
Force	Pinch Force	>15 N	>25 N	29 N
	Transmission in comparisson to 100-Dollar Hand	>90%	-	95 %
Control system	Power	6 Volt	-	6 Volt
	Lock	Locking capabilities	-	Locking capabilities
Weight	Weight hand	< 400 gram	< 300 gram	320 gram
Costs	Material costs	< 200 dollar	-	< 200 dollar

8.6. Future research

The developed prototype has some promising results in terms of the amplified pinch force and accuracy of control. However, some challenges remain. For the prototype the worm gear efficiency could not be increased beyond ∼25%. Due to low efficiency a motor with a higher transmission ratio had to be used. Furthermore the stalling torque of the motor was around 50% lower than stated by the supplier. With the higher transmission ratio comes a lower output speed (RPM). From the original design the motor could theoretically close the hand in approximately 0.5 seconds. In the final prototype the closing time of the motor (without user input) was increased to 3.5 seconds. Because the motor base crank is connected by a spring to the user base crank this closing time can be faster in combination with user input. The user should in this case overcome the spring forces just like the 100-Dollar Hand, which had a spring with a lower spring constant. For future research a precision worm gear could be used to increase the efficiency to around 80%, this would be sufficient to use the initial motor specifications. Another option to look into is a method of decoupling and coupling the spring to optimize the functionality of the prosthesis. By decoupling the spring when the user wants to close the hand and coupling the spring when the motor is amplifying the pinch force, the user will have more advantages of a body powered prosthesis.

A custom made spring which has a slightly lower constant (10-15 N/mm) and still be able to store over 150 N could make it easier for the user to overcome the spring forces to close the hand. This would also make it easier for the motor to take smaller steps in terms of amplified pinch force (increase resolution). For the control system an Arduino Nano Every has been used, this is the smallest micro controller available from Arduino. To fit the control system into the outer shell of the hand, research could be done into other available micro controllers. A custom made circuit board could make the control system less bulky. To measure input forces of the system a Miniature S-Beam Jr. Load Cell from FUTEK was available for use. The load cell has dimensions of 19.1x17.5x4.7 mm. With the shell of the hand only being 7 cm long the sensor does not fit into the shell. A smaller (maybe less accurate) load cell could allow the sensor to be fitted into the hand.

9. Conclusion

This study presents a prosthetic hand with force assist, the Hybrid Hand. The goal of this study was to design and validate a prosthetic hand with force assist, that allows for user operation without muscle fatigue, while still providing the user with transparent feedback. The developed prototype was able to amplify the actuation forces of the user. The amplification was done in such a manner, that the relation between the input actuation force and output pinch force was kept linear. To achieve a fatigue free operation a pinch force of 15 N has to be realised for an actuation force of 38 N. Our designed prototype was able to deliver 15 N pinch force with an actuation force of 30 N, thus able to be operated fatigue free. Nevertheless, some improvements could be made to the system with future research. Due to the low efficiency of the worm gear, we had to give in on the speed of the actuation (3.5 seconds to close the hand). Future research should be aimed towards increasing the efficiency of the transmission and towards a motor with more power. In addition a custom made PCB or micro controller allows the control system to be integrated in the hand. Making the Hybrid Hand respond faster, stronger and easier to wear.

References

[1] D. Plettenburg, Upper extremity prosthetics: current status & evaluation. 2006, Delft The Netherlands.

[2] F. Cordella, A. L. Ciancio, R. Sacchetti, A. Davalli, A. G. Cutti, E. Guglielmelli, L. Zollo, Literature review on needs of upper limb prosthesis users, Frontiers in neuroscience 10 (2016) 209.

[3] B. Maat, G. Smit, D. Plettenburg, P. Breedveld, Passive prosthetic hands and tools: A literature review, Prosthetics and orthotics international 42 (1) (2018) 66–74.

[4] R. Ayub, D. Villarreal, R. D. Gregg, F. Gao, Evaluation of transradial body-powered prostheses using a robotic simulator, Prosthetics and Orthotics International 41 (2) (2017) 194–200.

[5] G. Smit, D. H. Plettenburg, Efficiency of voluntary closing hand and hook prostheses, Prosthetics and orthotics international 34 (4) (2010) 411–427.

[6] L. Jones, J. Davidson, Save that arm: a study of problems in the remaining arm of unilateral upper limb amputees, Prosthetics and orthotics international 23 (1) (1999) 55–58.

[7] E. A. Biddiss, T. T. Chau, Upper limb prosthesis use and abandonment: a survey of the last 25 years, Prosthetics and orthotics international 31 (3) (2007) 236–257.

[8] M. Hichert, A. N. Vardy, D. Plettenburg, Fatigue-free operation of most body-powered prostheses not feasible for majority of users with trans-radial deficiency, Prosthetics and orthotics international 42 (1) (2018) 84–92.

[9] P. Svensson, U. Wijk, A. Björkman, C. Antfolk, A review of invasive and non-invasive sensory feedback in upper limb prostheses, Expert review of medical devices 14 (6) (2017) 439–447.

[10] D. Farina, S. Amsüss, Reflections on the present and future of upper limb prostheses, Expert review of medical devices 13 (4) (2016) 321–324.

[11] S. L. Carey, D. J. Lura, M. J. Highsmith, Differences in myoelectric and body-powered upper-limb prostheses: Systematic literature review., Journal of Rehabilitation Research & Development 52 (3) (2015).

[12] A. Calado, F. Soares, D. Matos, A review on commercially available anthropomorphic myoelectric prosthetic hands, pattern-recognition-based microcontrollers and semg sensors used for prosthetic control, in: 2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), IEEE, 2019, pp. 1–6.

[13] P. Beckerle, S. Willwacher, M. Liarokapis, M. P. Bowers, M. B. Popovic, Prosthetic limbs, Biomechatronics (2019) 235.

[14] M. B. Popović, Biomechanics and robotics, CRC Press, 2013.

[15] G. Smit, The hundred dollar hand, a 3d-printed prosthesis for developing countries, in: Proc. ISPO 16th world congress, ISPO, 2019.

[16] M. Hichert, D. A. Abbink, A. N. Vardy, C. K. van der Sluis, W. G. Janssen, M. A. Brouwers, D. H. Plettenburg, Perception and control of low cable operation forces in voluntary closing body-powered upper-limb prostheses, PloS one 14 (11) (2019) e0225263.

[17] C. L. Taylor, The biomechanics of the normal and of the amputated upper extremity, Human limbs and their substitutes. New York: McGraw-Hill (1954) 169–221.

[18] N. Smaby, B. Baker, M. Johanson, J. Towles, W. Murray, Determination of lateral pinch force requirements for six common activities of daily living, in: Proceedings of the Third National Meeting for Rehabilitation Research and Development Conference, 2002, p. 104.

[19] N. Smaby, M. E. Johanson, B. Baker, D. E. Kenney, W. M. Murray, V. R. Hentz, Identification of key pinch forces required to complete functional tasks., Journal of Rehabilitation Research & Development 41 (2) (2004).

[20] Arduino, Arduino nano every (2020).
URL `https://store.arduino.cc/arduino-nano-every`

[21] G. Smit, R. M. Bongers, C. K. Van der Sluis, D. H. Plettenburg, Efficiency of voluntary opening hand and hook prosthetic devices: 24 years of development, J Rehabil Res Dev 49 (4) (2012) 523–34.

[22] A. Van Boeijen, J. Daalhuizen, R. van der Schoor, J. Zijlstra, Delft design guide: Design strategies and methods, 2014.

[23] S. Tromp, The design of a fully 3d printed easy-to-assemble and highly functional body-powered hand prosthesis for low-resource countries (2020).

[24] R. Guerra, I. Fonseca, F. Pichel, M. Restivo, T. Amaral, Hand length as an alternative measurement of height, European journal of clinical nutrition 68 (2) (2014) 229–233.

[25] P. Grasgruber, M. Sebera, E. Hrazdíra, J. Cacek, T. Kalina, Major correlates of male height: A study of 105 countries, Economics & Human Biology 21 (2016) 172–195.

[26] J. Liu, H. Peng, Z. Filipi, Modeling and analysis of the toyota hybrid system, TIc 200 (3) (2005).

[27] E. A. Casteel, M. Archibald, A study on the efficiency of bicycle hub gears, in: ASME International Mechanical Engineering Congress and Exposition, Vol. 56420, American Society of Mechanical Engineers, 2013, p. V013T14A044.

[28] J. W. Sensinger, R. F. F. Weir, Improvements to series elastic actuators, in: 2006 2nd IEEE/ASME International Conference on Mechatronics and Embedded Systems and Applications, IEEE, 2006, pp. 1–7.

[29] Pololu Corporation, Micro metal gearmotors (2020).
URL `https://www.pololu.com/category/60/micro-metal-gearmotors`

[30] E. F. Kececi, Mechatronic Components: Roadmap to Design, Butterworth-Heinemann, 2018.

[31] FUTEK, Miniatur s-beam jr. load cell (2020).
URL `https://www.futek.com/store/legacy-sensors-and-instruments/miniature-s-beam-LSB200/FSH00102`

[32] TEVEMA Technical Springs, Torsion spring catalogue (2020).
URL `https://www.tevema.com/nl/webshop/trekveren/trekveren`

[33] Mädler, Worm gears (2020).
URL `https://www.maedler.de/product/1643/1619/705/1931/1933/schneckenraeder-aus-bronze-eingaengig-rechts-modul-05`

[34] M. Turci, E. Ferramola, F. Bisanti, G. Giacomozzi, Worm gear efficiency estimation and optimization, Gear Technology 33 (4) (2016) 46–53.

[35] Ultimaker BV, Ultimaker 3 description (2019).
URL `https://ultimaker.com/nl/3d-printers/ultimaker-3`

[36] Tinkercad, Arduino simulator (2020).
URL `https://www.tinkercad.com`

Appendix A. Free Body Diagrams

For the mechanical analyses of the closing mechanism a free body diagram has been made for each of the linkages. With the direction of forces known the static equations can be derived. For all of the situations the sum of all forces in the x-direction ($\Sigma F_x = 0$) should be 0 and the sum of all forces in the y-direction should be 0 ($\Sigma F_y = 0$).

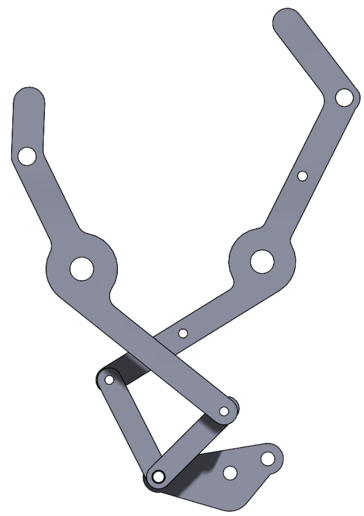


Figure A.1: Closing mechanism with all of the linkages

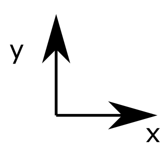


Figure A.2: Direction of the x-y axis

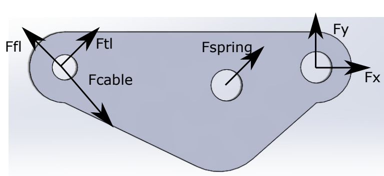


Figure A.3: Free body diagram of the base link

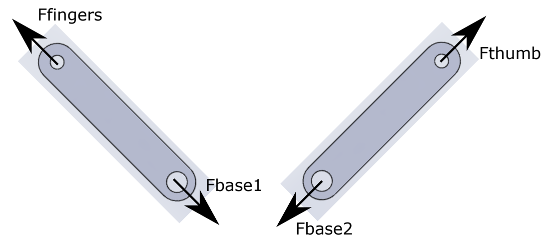


Figure A.4: Free body diagrams of the 'floating' thumb and finger link

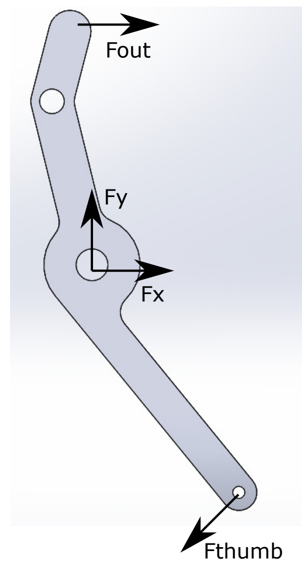


Figure A.5: Free body diagram of the thumb link

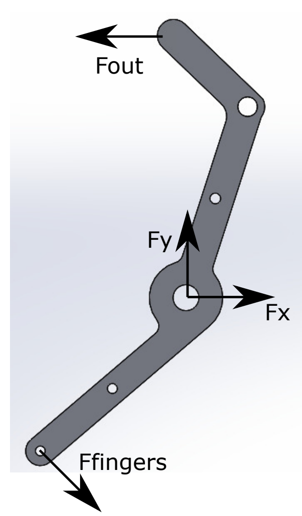


Figure A.6: Free body diagram of the finger link

Appendix B. Forces floating links

The forces of the two floating links can be seen in Figure B.1 over the closing of the hand with an input actuation force of 100 N. One thing to notice is that the sum of both forces can exceed 100 N due to the angle of the forces of the floating links.

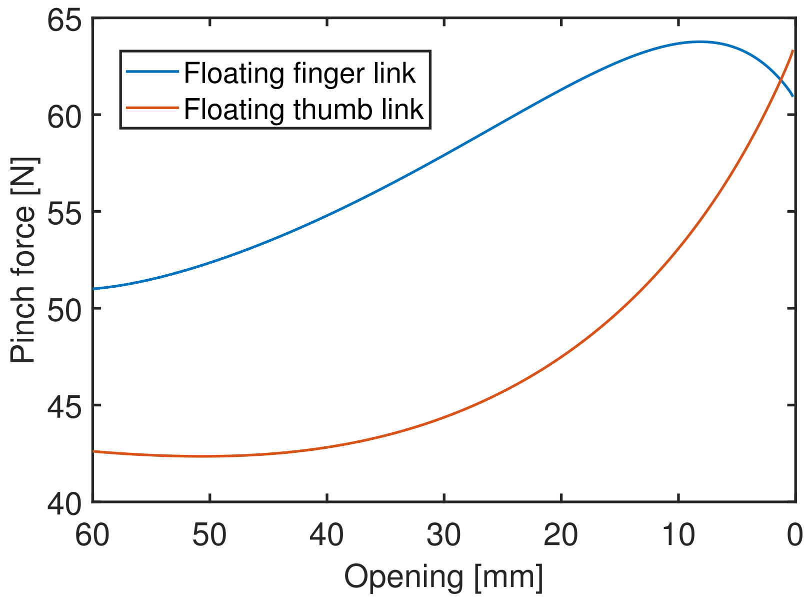


Figure B.1: Forces of the two floating links

Appendix C. Matlab model Closing Mechanism

Matlab model of the mechanical analyses of the closing mechanism of the 100-Dollar Hand. In this model parameters that can be changed are: lengths of the cranks, coordinates of the rotation points and input force. In this script first the coordinates of all of the cranks in each of the (resolution) situations are calculated using the cosine rule. After that, the forces are calculated for each of these situations. For the angles between the vectors the dot product is used. In the next section the closing mechanism is plotted. Lastly, the model is validated with the tensile tests of the 100-Dollar Hand.

Luuk Lommerse 14-2

Opening and closing of the 100 dollar hand

In this script the Variables and System points can be changed to change the closing mechanism

```
%Close and clear all previous values
clc
close all
clear all
```

Modelpar

```
%Make 1 to plot system, 0 to skip plotting
Draw=0;
%0(off), 1, 2 or 3 (compares actual
pull/pinchtest data with model)
Validatietest=1;
```

Variables

```
%Number of situations calculated, 1.000 by
default
Resolutie=1000;
%user input force
Fin=100;  %N
lrust_veer=21.7; %mm veer
Kveer=.92;   %N/mm
Voorspanning=2.6; %N

l1=24.75;   %Length crank 1(mm) (point a to
point 1)
l2=23;  %Length crank 2(mm) (point 1 to
point 2 (thumbpart))
l3=27; %Length crank 3(mm) (point 1 to
point 3 (fingerpart))
l4=43.2;    %Length crank 4(mm) (point 3 to
point C (fingers))
l5=45;  %Length crank 5(mm) (point2 to
point B (Thumb))
l6=64;  %Length crank 6(mm) (point C to FT
(finger(tip))
l7=51.8;       %Lenght crank 7(mm) (point B
to TT (Thumb (tip))
anglett=deg2rad(144.9); %Angle of the thumb
crank with thumb tip
angleft=deg2rad(107.25); %Angle of the
```

System points

```
%Rotation point A
xa=0;                       %x
coordinate(s) rotation point a
ya=0;                        %y
coordinate(s) rotation point a
%Rotation point B (Thumb)
xb=xa-41.587*sf;                    %x
coordinate(s) rotation point B
yb=ya+42.444*sf;                 %x
coordinate(s) rotation point B
%rotation Point C (Fingers)
xc=xa-1.25*sf;                 %x
coordinate(s) rotation point C
yc=xa+44.064*sf;                 %y
coordinate(s) rotation point C

%Out bowden cable
xforce=xa+4.93;
%xcoordinaat output draad
yforce=ya-33.42;
%ycoordinaat output draad

% Determine vector from max open to max
close
% Input vector rho1
%-.157*pi = 60.02mm open
rho1=(-.157*pi:-pi/Resolutie:-.508*pi);
n=length(rho1);
```

4bar linkage Thumb

```
d=sqrt(xb^2 +yb^2);    %Determine Length
base (4th bar)

E=sqrt(l1^2+d^2-2*l1*d*cos(rho1));
%Help-line to calculate angles (Cos rule)
alpha=asin((l1*sin(rho1))./E);
%First part angle bar 5 with vertical
beta=acos((E.^2+l5^2-l2^2)./(2*E*l5));
%Second part angle bar 5 with vertical
gamma=atan(abs(xb/yb));
%Third (last) part angle bar 5 with
vertical
zeta=atan(abs(yb/xb));
%2nd Help-line with horizontal

%Point 1
x1=l1*cos(pi-zeta-rho1);
y1=l1*sin(pi-zeta-rho1);

%point 2
x2=xb+l5*sin((alpha+beta+gamma));
y2=yb-l5*cos((alpha+beta+gamma));
```

```
%crank4 thumb
thetat=anglett-(1/2*pi-(alpha+beta+gamma));
xtt=xb+l7*cos(thetat);
ytt=yb+l7*sin(thetat);
```

4 bar linkage Fingers

Finger

```
q=atan(abs(yc/xc));
rho2=pi-(zeta+rho1+q);
df=sqrt(xc^2 +yc^2);
Ef=sqrt(l1^2+df^2-2*l1*df*cos(rho2));
alphaf=asin((l1*sin(rho2))./Ef);
betaf=acos((Ef.^2+l4^2-l3^2)./(2*Ef*l4));
gammaf=atan(abs(xc/yc));

x3=xc-l4*sin(alphaf+betaf+gammaf);
y3=yc-l4*cos(alphaf+betaf+gammaf);

%crank7finger
thetaf=anglеft-(1/2*pi-
(alphaf+betaf+gammaf));
 xft=xc-l6*cos(thetaf);
 yft=yc+l6*sin(thetaf);
```

Force calculations

```
xr=abs(x1)+abs(xforce);           %x rope
output and x1
yr=(y1)+abs(yforce);           %y rope
output and y1
L1_r=sqrt(xr.^2+yr.^2);   %Length x1 to
exit
La_r=sqrt(yforce^2+xforce^2); %length base
to exit

Angledraad=atan(abs(xforce-x1)./abs(yforce-
y1));   %hoek van het draad
F_in_ver=Fin.*cos(Angledraad);
%Verticaalkracht

Momenthoek=acos((l1.^2+L1_r.^2-
La_r.^2)./(2*l1.*L1_r));
MomentA=Fin.*cos(1/2*pi-Momenthoek).*l1;

%veer
xveer1=xa+2.086;
yveer1=ya+27.144;

%Point veer
xveer2=9*cos(pi-zeta-rho1);
yveer2=9*sin(pi-zeta-rho1);
l_veer=(sqrt((xveer2-xveer1).^2+(yveer2-
yveer1).^2));
uitrekking_veer=sqrt((xveer2-
xveer1).^2+(yveer2-yveer1).^2)-lrust_veer;

RVveer1=[0;0]-[xveer2;yveer2];
RVveer2=[xveer1;yveer1]-[xveer2;yveer2];
Hoekveer=acos(sum(RVveer1.*RVveer2)./((sqrt
(sum(RVveer1.^2))).*sqrt(sum(RVveer2.^2))))
;
Hoekveer1=abs(1/2*pi-Hoekveer);
Mveer=(uitrekking_veer*Kveer+Voorspanning).
*cos(Hoekveer1)*9;
%Moment around A
Ma=MomentA-Mveer;

Afstand=sqrt((xft-xtt).^2+(yft-ytt).^2);
Opening=sqrt((xft-xtt).^2);
Rope_dist=sqrt((x1(1)-
x1(1:end)).^2+((y1(1)-y1(1:end))).^2);

PuntFT=[xft;yft];               %vector met
x-y coordinaten van fingertip
PuntTT=[xtt;ytt];               %vector met
x-y coordinaten van thumbtip

Richtingsvector=PuntFT-PuntTT;
RV_FT=PuntFT-[xc;yc];
RV_TT=PuntTT-[xb;yb];
```

```
%Hoek van richtingsvector force met
richtingsvector mechaniek FT/TT
HoekFT_richtingsvector=acos(sum(Richtingsve
ctor.*RV_FT)./((sqrt(sum(Richtingsvector.^2
))).*sqrt(sum(RV_FT.^2))));
HoekTT_richtingsvector=acos(sum(Richtingsve
ctor.*RV_TT)./((sqrt(sum(Richtingsvector.^2
))).*sqrt(sum(RV_TT.^2))));

HoekFThor=abs(1/2*pi-
HoekFT_richtingsvector); %Hoek kracht met
horizontaal
HoekTThor=abs(1/2*pi-
HoekTT_richtingsvector);%Hoek kracht met
horizontaal

%Link en tip mechaniek
RV_FL_FT=[x1;y1]-[x3;y3];
RV_TL_TT=[x1;y1]-[x2;y2];
RV_FL_x=[xc;yc]-[x3;y3];
RV_TL_x=[xb;yb]-[x2;y2];

%Hoek van richtingsvector force met
richtingsvector mechaniek FT/TT
HoekRVFL=acos(sum(RV_FL_x.*RV_FL_FT)./((sqr
t(sum(RV_FL_x.^2))).*sqrt(sum(RV_FL_FT.^2))
));
HoekRVTL=acos(sum(RV_TL_x.*RV_TL_TT)./((sqr
t(sum(RV_TL_x.^2))).*sqrt(sum(RV_TL_TT.^2))
));
```

```
HoekFT_tan=abs(1/2*pi-HoekRVFL); %Hoek
kracht met tangentiaal
HoekTT_tan=abs(1/2*pi-HoekRVTL);%Hoek
kracht met tangentiaal

RatioFTout=l6./(l4*cos(HoekFT_tan));
RatioTTout=l7./(l5*cos(HoekTT_tan));
Ratio=RatioFTout./RatioTTout;
%Link FT force = x * Link TT force

RVbase=[xa;ya]-[x1;y1];
RVTL=[x2;y2]-[x1;y1];
RVFL=[x3;y3]-[x1;y1];

%hoek met base
HoekFL_base=acos(sum(RVbase.*RVFL)./((sqrt(
sum(RVbase.^2))).*sqrt(sum(RVFL.^2))));
HoekTL_base=acos(sum(RVbase.*RVTL)./((sqrt(
sum(RVbase.^2))).*sqrt(sum(RVTL.^2))));

HoekFL_tan=abs(1/2*pi-HoekFL_base);
HoekTL_tan=abs(1/2*pi-HoekTL_base);

FLT=(Ma/l1)./(cos(HoekTL_tan)+Ratio.*cos(Ho
ekFL_tan));
FLF=FLT.*Ratio;

FoutFT=FLF./RatioFTout;
FoutTT=FLT./RatioTTout;

Fout=FoutTT;
```

Plot

```
if Draw==1

hp=figure('Position',[10 10 1000 1000]);
plot(0,0,'ok','MarkerSize',15,'LineWidth',2
)
hold on
attachment=plot(xb,yb,'ok','MarkerSize',15,
'LineWidth',2)
plot(xc,yc,'ok','MarkerSize',15,'LineWidth'
,2)
Path=plot(x1,y1,'r','MarkerSize',15,'LineWi
dth',2)
plot(xveer1,yveer1,'ok','MarkerSize',15,'Li
neWidth',2)
xlim([-80 max(xft)+50])
ylim([(-20) (max(yft)+45)])

while ishandle(hp)
    for i=1:n
%       i=floor(n/2)

crank1=line([xa,x1(i)],[ya,y1(i)],'LineWidt
h',2);

crank2=line([x1(i),x2(i)],[y1(i),y2(i)],'Li
neWidth',2);

crank3=line([x2(i),xb],[y2(i),yb],'LineWidt
h',2);

crank4=line([xb,xtt(i)],[yb,ytt(i)],'LineWi
dth',2);

crank5=line([x1(i),x3(i)],[y1(i),y3(i)],'Li
neWidth',2);

crank6=line([x3(i),xc],[y3(i),yc],'LineWidt
h',2);


crank7=line([xc,xft(i)],[yc,yft(i)],'LineWi
dth',2);

spring1=line([xveer1,xveer2(i)],[yveer1,yve
er2(i)],'Color',[0.4660 0.6740
0.1880],'LineStyle','--','LineWidth',2);

Floating=plot(x1(i),y1(i),'*k','MarkerSize'
,25)

Floating=plot(x2(i),y2(i),'*k','MarkerSize'
,25)

Floating=plot(x3(i),y3(i),'*k','MarkerSize'
,25)
       plot(0,130)
       set(gca,'XTick',[], 'YTick', [])
           lgd=legend([attachment Floating
crank1 spring1 Path],' Rotary point
attached to the shell of the hand', '
Floating rotary point', ' Crank',' Spring',
' Range of motion')
        lgd.FontSize = 20;
        lgd.FontWeight = 'bold';
            pause(0.01)
         delete(crank1)
         delete(crank2)
         delete(crank3)
      delete(crank4)
      delete(crank5)
       delete(crank6)
       delete(crank7)
       delete(spring1)
      if ~ishandle(hp)
             break;
             return
      end

    end
```

```
        for j=length(x1):-1:1
        if ~ishandle(hp)
             break;
        end
        crank1=line([xa,x1(j)],[ya,y1(j)]);

crank2=line([x1(j),x2(j)],[y1(j),y2(j)]);
        crank3=line([x2(j),xb],[y2(j),yb]);
      crank4=line([xb,xtt(j)],[yb,ytt(j)]);

crank5=line([x1(j),x3(j)],[y1(j),y3(j)]);
       crank6=line([x3(j),xc],[y3(j),yc]);

crank7=line([xc,xft(j)],[yc,yft(j)]);

spring1=line([xveer1,xveer2(j)],[yveer1,yve
er2(j)],'Color',[0.4660 0.6740
0.1880],'LineStyle','--');
        pause(0.01)
         delete(crank1)
        delete(crank2)
         delete(crank3)
        delete(crank4)
      delete(crank5)
       delete(crank6)
       delete(crank7)
       delete(spring1)
    end
end
end
```

Validation

```
if Validatietest==1 || Validatietest==2 ||
Validatietest==3
warning('off','MATLAB:table:ModifiedAndSave
dVarnames')
% Read pull test results
switch Validatietest
    case 1
        Test=readtable('100D_Pull1.txt');
    case 2
        Test=readtable('100D_Pull2.txt');
    case 3
        Test=readtable('100D_Pull3.txt');
end

%Pinchtest

PullforceN=table2array(Test(:,8));
PinchforceN=table2array(Test(:,11));

figure
plot(PullforceN);
hold on
plot(PinchforceN);
```

```
%find value for pinchforcesensor (10mm)
[~,Searchingindex]=min(abs(Afstand-10));
S_index2=find(PinchforceN >= .2, 1,
'first');
S_index3=find(PinchforceN >= .2, 1,
'last');

% S_index3=S_index3+S_index2+9;
%calculate moments om A
ValMa=-
PullforceN(S_index2:S_index3)'.*cos(1/2*pi-
Momenthoek(Searchingindex)).*l1-
Mveer(Searchingindex);

FLT_val=(ValMa/l1)./(cos(HoekTL_tan(Searchi
ngindex))+Ratio(Searchingindex).*cos(HoekFL
_tan(Searchingindex)));
FLF_val=FLT_val.*Ratio(Searchingindex);

FoutFT_val=FLF_val./RatioFTout(Searchingind
ex);

[~,plotindex]=min(PullforceN);
figure
plot(FoutFT_val)
hold on
plot(PinchforceN(S_index2:S_index3))
legend('Model','Trektest')

%Travel distance rope
Dist_rope_test=table2array(Test(:,9));
ValRope_dist_error=Dist_rope_test(S_index2)
-Rope_dist(Searchingindex)-3

figure
plot(abs(PullforceN(S_index2:S_index3)),Fou
tFT_val)
hold on
plot(abs(PullforceN(S_index2:plotindex)),mo
vmean((PinchforceN(S_index2:plotindex)),5))
end
```

Extra plot

```
%plot 100N opening-pinchforce
figure
plot(Afstand,Fout,'LineWidth',2)
xlim([0 60])
set ( gca, 'xdir', 'reverse' )
ylabel('Pinchforce [N]')
xlabel('Opening [mm]')

%Tensile test
figure
plot(FoutFT_val,'LineWidth',2)
hold on
plot(PinchforceN(S_index2:S_index3),'LineWi
```

```
dth',2)
legend('Model','Trektest')
```

import tensile test data

```
Test1 = load('Tensiletest1.mat');
[~,plotindex1]=min(Test1.PullforceN);
Test2 = load('Tensiletest2.mat');
[~,plotindex2]=min(Test2.PullforceN);
Test3 = load('Tensiletest3.mat');
[~,plotindex3]=min(Test3.PullforceN);
Movfilter=1;

figure
%plot model
plot(abs(PullforceN(S_index2:S_index3)),Fou
tFT_val,'LineWidth',2)
hold on
%plot Test1
plot(abs(Test1.PullforceN(Test1.S_index2:pl
otindex1)),movmean((Test1.PinchforceN(Test1
.S_index2:plotindex1)),Movfilter),'r','Mark
erSize',10,'LineWidth',4)
%plot Test2
plot(abs(Test2.PullforceN(Test2.S_index2:pl
otindex2)),movmean((Test2.PinchforceN(Test2
.S_index2:plotindex2)),Movfilter),'k','Mark
erSize',10,'LineWidth',3)
%plot Test3
plot(abs(Test3.PullforceN(Test3.S_index2:pl
otindex3)),movmean((Test3.PinchforceN(Test3
.S_index2:plotindex3)),Movfilter),'y','Mark
erSize',10,'LineWidth',2)

ylabel('Pinchforce [N]')
xlabel('Actuation Force [N]')
legend('Model','Tensile pinchforce
test1','Tensile pinchforce test 2','Tensile
pinchforce test 3')
open('Tensiletest1.mat')
```

Plot mechanism

```
hp=figure('Position',[10 10 1000 1000]);
plot(0,0,'ok','MarkerSize',15,'LineWidth',2
)
hold on
attachment=plot(xb,yb,'ok','MarkerSize',15,
'LineWidth',2)
plot(xc,yc,'ok','MarkerSize',15,'LineWidth'
,2)
Path=plot(x1,y1,'r','MarkerSize',15,'LineWi
dth',2)
plot(xveer1,yveer1,'ok','MarkerSize',15,'Li
neWidth',2)
xlim([-80 max(xft)+50])
ylim([(-20) (max(yft)+45)])
```

```
    i=floor(n/2)

crank1=line([xa,x1(i)],[ya,y1(i)],'LineWidt
h',2);

crank2=line([x1(i),x2(i)],[y1(i),y2(i)],'Li
neWidth',2);

crank3=line([x2(i),xb],[y2(i),yb],'LineWidt
h',2);

crank4=line([xb,xtt(i)],[yb,ytt(i)],'LineWi
dth',2);

crank5=line([x1(i),x3(i)],[y1(i),y3(i)],'Li
neWidth',2);

crank6=line([x3(i),xc],[y3(i),yc],'LineWidt
h',2);


crank7=line([xc,xft(i)],[yc,yft(i)],'LineWi
dth',2);

spring1=line([xveer1,xveer2(i)],[yveer1,yve
er2(i)],'Color',[0.4660 0.6740
0.1880],'LineStyle','--','LineWidth',2);

Floating=plot(x1(i),y1(i),'*k','MarkerSize'
,25)

Floating=plot(x2(i),y2(i),'*k','MarkerSize'
,25)

Floating=plot(x3(i),y3(i),'*k','MarkerSize'
,25)
       plot(0,130)
       set(gca,'XTick',[], 'YTick', [])
           lgd=legend([attachment Floating
crank1 spring1 Path],' Rotary point
attached to the shell of the hand', '
Floating rotary point', ' Crank',' Spring',
' Range of motion')
 lgd.FontSize = 20;
lgd.FontWeight = 'bold';
```

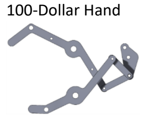
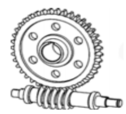
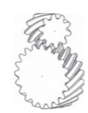
```
ylabel('Pinchforce [N]')
xlabel('Opening [mm]')
```

Published with MATLAB® R2017b

Appendix D. Morphologic overview and concepts

In Table D.1 solutions have been summarized for each of the components. In Figures D.1-D.8 conceptual solutions for the addition of two forces (hybrid mode) have been shown.

Table D.1: Morphologic overview for the separate components

Component	**Solution 1**	**Solution 2**	**Solution 3**
Outer shell	Shell 100-Dollar Hand 	New shell	
Closing mechanism	Closing mechanism 100-Dollar Hand 	New closing mechanism	
Actuator	Servo motor  [https://www.conrad.nl/]	Stepper motor  [https://www.conrad.nl/]	DC Motor  [https://www.pololu.com/]
Transmission	Worm gear  [30]	Bevel gears  [30]	Crossed helical gears  [30]
Springs	Tension spring  [https://www.tevema.com/nl/]	Torsion spring  [https://www.tevema.com/nl/]	

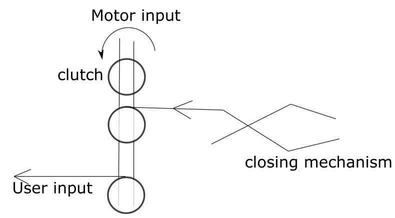


Figure D.1: Concept 1: Actuator input coupled with a one-way clutch to the user axis. This allows the user to overrule the system in terms of closing the hand. For opening the hand the actuator should be used.

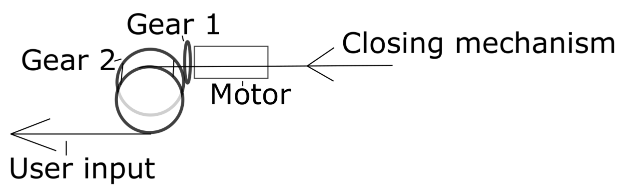


Figure D.2: Concept 2: Actuator drives a pulley (with gear transmission) which is directly connected to a pulley of the user input.

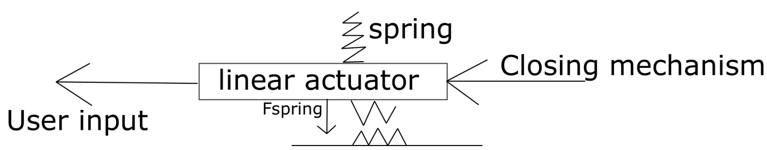


Figure D.3: Concept 3: Linear actuator which grips into teeth of a gear when spring force presses the actuator down. When gripped to the shell the linear actuator can amplify the pinch force.

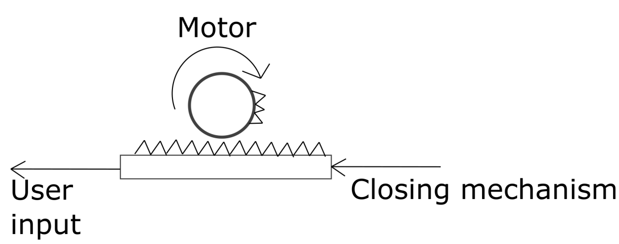


Figure D.4: Concept 4: An element with a linear gear is moving along a rotation point of the motor when the user actuates the cable. By actuating the motor the pinch force can be amplified. The teeth of the motor pulley grip into the element of the user input.

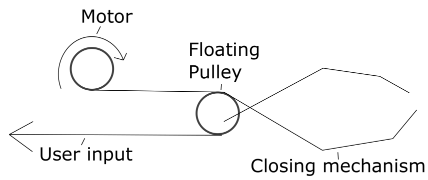


Figure D.5: Concept 5: The user actuates a floating pulley which can move horizontally. To amplify the pinch force the motor drives the pulley from the other end of the rope. By changing the closing mechanism a rotation pulley could also be viable.

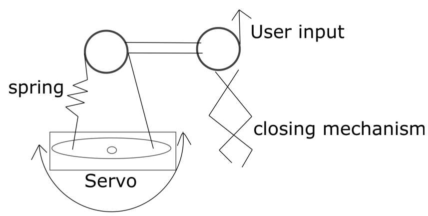


Figure D.6: Concept 6: Series elastic actuator, a servo actuator is attached with a spring and a cable to 2 sides of a pulley. The pulley is connected to the user pulley and input with a shaft (possibly a clutch). By controlling the servo's position the spring will be tensioned and adds forces to the user input.

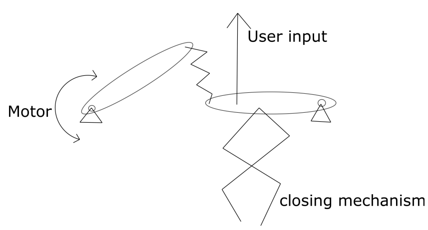


Figure D.7: Concept 7: Series elastic actuator, by changing the output torque of the motor the spring can be tensioned both ways. This allows for control of an added moment force for closing and opening the hand.

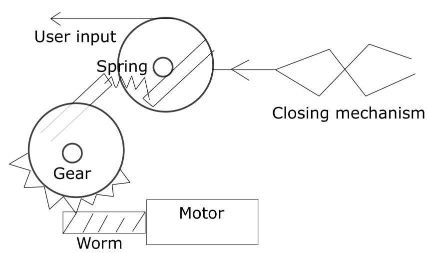


Figure D.8: Concept 8: Final concept, by controlling the position of the gear on the motor side, the spring can be tensioned to amplify the forces of the user to open or close the hand. When using a worm gear as transmission, the motor can stop when the desired position is reached.

Appendix E. Pinch force and Motor analyses

The maximal pinch force the system is able to deliver is much lower than expected. In this Appendix we explain the steps we took to optimize the system and transmission. For clarity purposes, the figures have been added to the next page. Every component in the hand has been analysed to find the problem. Motor with 1:50 transmission led to 3 N pinch force and 1:150 led to 6.5 N which should be 29 N following the calculations. To find the problem following steps have been taken for the system:

1. The calculations have been made by hand and checked with Matlab solutions.
2. Frictions in the system have been checked
   (a) This led to an increase of the length of the motor base crank (to 14 mm) to avoid high pre-tensioning of the spring.
   (b) Potentiometer has been aligned better.
   (c) Aligned the closing mechanisms, less losses by forces in the z-direction (perpendicular).
   (d) Aligned the spring, which leads to less force horizontally (Figure E.1)

After the friction check the torque of the motor has been tested. The steps that have been taken for the motor tests are:

1. Motor test with weights, a pulley of 1 cm radius has been 3D printed (Figure E.2).
   (a) For the 1:150 transmission: Supplier motor values: 2.4 kg*cm stall torque, measured 1-1.5 kg*cm
   (b) With 1 kg*cm stall torque the pinch force should be 17 N following calculations.
2. The motor torque after the worm gear transmission has been tested (Force hypothesis: (1kg*cm * 20(transmission) * 0.53(efficiency)) / 1.4(cm arm) =7.6 kg.
   (a) Shaft of the motor seems to be bending, this is causing the worm to slip on the gear(Result 2 kg elevated).
   (b) An extra bearing has been added in extension of the motor shaft (Figure E.3) (Result 2.5 kg elevated).
   (c) Add an extra bearing to the extension of the motor shaft. This to exclude bearing failure under high radial forces (2.5 kg elevated).
3. Video analyses of the movement of the motor shaft show movement (in the direction of the motor).
4. Axial forces have been checked.
   (a) Resulted in decrease in motor speed (and increase of Ampere) with an axial load of 20 N.
5. The (movement of the) motor gearbox has been checked.
   (a) Due to the axial load one of the gears is pushed against the housing, resulting in high friction (Figure E.4).
   (b) An axial bearing has been added, where the big ring is touching the housing and the small ring the worm gear (Figure E.5). This resulted in less speed reduction and a lower Amperage (.2 instead of .45).
6. Strength of the motor bracket has been increased, an extra internal structure has been added (Figure. The axial bearing touches the bracket which decreases the resulting radial force on the motor shaft.

Figures E.1-6 can be found on the next page.

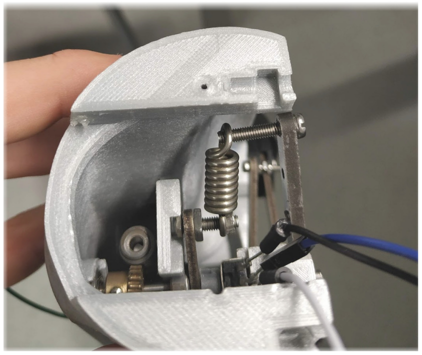


Figure E.1: Spring has been aligned so that the horizontal forces are minimal.

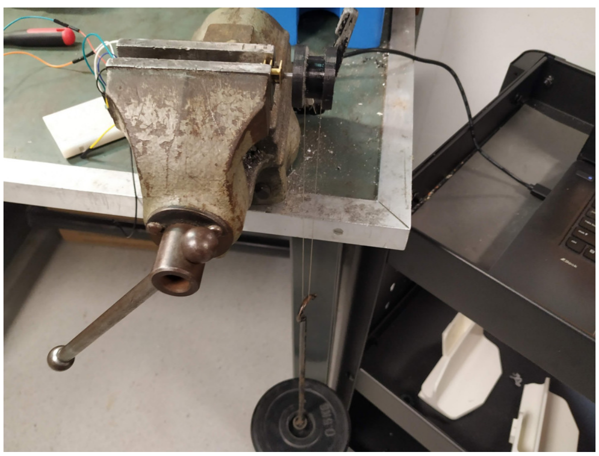


Figure E.2: A pulley has been attached to the motor shaft. The weights are elevated on a 1 cm radius

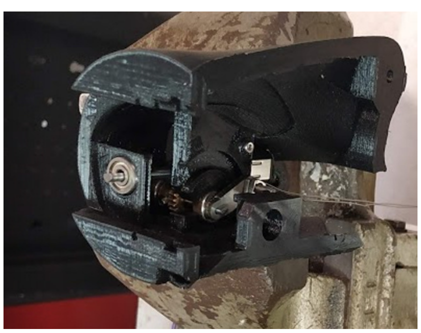


Figure E.3: A bearing has been added in the extension of the motor shaft, this prevents the worm from slipping on the gear

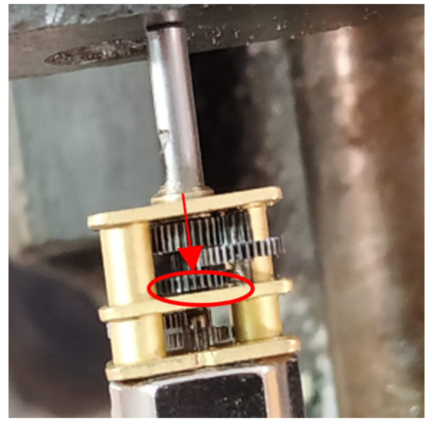


Figure E.4: Due to the axial load one of the gears is pushed against the housing, resulting in high friction.

Figure E.5: Due to the axial load one of the gears is pushed against the housing, resulting in high friction.

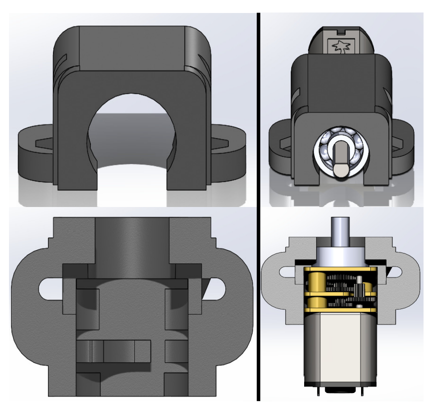


Figure E.6: Left: Designed motor bracket with inner blocks Right: Assembly of the motor bracket and the motor. The round shape at the front of the bracket allows the axial bearing to also refrain the motor shaft from bending due to radial forces.(Duplicate of Figure 9 for clarity reasons)

Appendix F. Potentiometer caps

Renders of the Solidwork models of the potentiometer caps. In Figure F.1-2 the first cap is visible. The hollow bottom side of the cap is attached to the shaft of the motor side potentiometer. Due to the slotted top of this cap the potentiometer with cap can be slide over the gear shaft into an enclosed area (3 sides). In Figure F.3-4 the second potentiometer cap can be seen. This cap fits on the potentiometer of the human side. A d-shaped hole can be seen in the top of the cap. This shape fits exactly over the shaft of the user base crank.

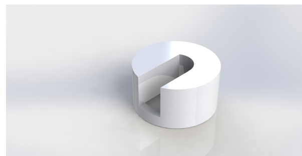


Figure F.1: View from above of the potentiometer cap for the motor side potentiometer.

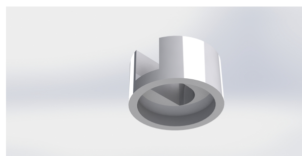


Figure F.2: View from below of the potentiometer cap for the motor side potentiometer.

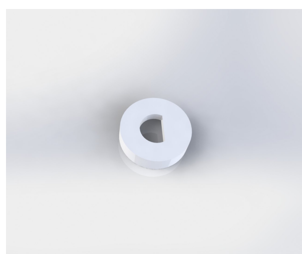


Figure F.3: View from above of the potentiometer cap for the user side potentiometer.

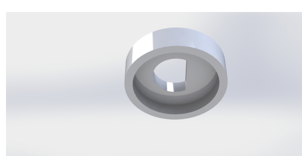


Figure F.4: View from below of the potentiometer cap for the user side potentiometer.

Appendix G. Matlab model Hybrid Hand

Matlab model for the Hybrid Hand. In this model the variables that can be changed are the radius of the user base crank, the spring characteristics and the springs in parallel. For the control system arrays are made with parameters for the function fit. These parameters are used when the user position is at the corresponding index of the array.

Luuk Lommerse 2020

Model script for the spring transmission

Values that can be changed are the variables and the spring characteristics

```
% Clear previous values
clear all
clc
```

Variables

```
Rhum=25;      %Radius human base crank
veer=4;       %Type of spring, see spring
types below
aantal_s=1;   %Number of springs in
parallel


for v=1:1
for h=1:1
```

```
R1=36-v;      %Humanside
R2=14+h;      %motorside
spx2=0;
Rendement=.25;  %wormgear
%case 1 3 4 options

R1=R1+1.5;      %Radius axis where spring
is attached on (1.5mm)
R2=R2-1.5;
```

Springs

```
switch veer
    case 1      %T42045
        l_spring_rust=23.6; %mm veer
        k=26.2*aantal_s;   %N/mm
        Voorspanning=26.65*aantal_s; %N
        max_l=6.69;
    case 2      %T41912
            l_spring_rust=24.8; %mm veer
            k=19.4*aantal_s;   %N/mm
            Voorspanning=20.64*aantal_s; %N
            max_l=6.8;
    case 3      %T31920
            l_spring_rust=22.1; %mm veer
            k=23.74*aantal_s;   %N/mm
            Voorspanning=24.36*aantal_s; %N
            max_l=6.75;

    case 4      %T41920
            l_spring_rust=22.1; %mm veer
            k=20.39*aantal_s;   %N/mm
            Voorspanning=20.84*aantal_s; %N
            max_l=6.76;
    case 5      %T31565
            l_spring_rust=19.4; %mm veer
            k=7.28*aantal_s;   %N/mm
            Voorspanning=9.85*aantal_s; %N
            max_l=9.47;
    case 6      %T31911
            l_spring_rust=17.3; %mm veer
            k=38.03*aantal_s;   %N/mm
            Voorspanning=24.15*aantal_s; %N
            max_l=3.99
    case 7      %T31913a
            l_spring_rust=24.3; %mm veer
            k=9.09*aantal_s;   %N/mm
            Voorspanning=14.84*aantal_s; %N
            max_l=11.56

end

for g=1:85 %Loop to function fit every
orientation of the cranks/springs
```

```
%128degrees open 45 degrees closed
0.6763*pi for closed pinchforce 0.26*pi
open, 0.7220*pi closed
Angle_1=deg2rad(130-g); %Human crank angle
Angle_2=(Angle_1+1/1000*pi:1/1000*pi:1*1.2*
pi); %Motor crank angle
```

Coordinates 1=human 2=motor

```
x1=R1*cos(Angle_1);
x2=-spx2+R2*cos(Angle_2);
y1=R1*sin(Angle_1);
y2=R2*sin(Angle_2);

%determine spring length if les than
resting lenght value is NaN
for i=1:length(Angle_2)
    l_spring(i)=sqrt((x1-x2(i)).^2+(y1-
y2(i)).^2);
     if l_spring(i)<l_spring_rust
         l_spring(i)=NaN;
     end
end

Angle_tan=atan(abs(y1-y2)./abs(x1-x2));
%tangential angle
F_spring=(l_spring-l_spring_rust)*k;
%Spring force

check=sqrt((x1-spx2)^2+(y1-0)^2);
%Make negatives positive
for j=1:length(Angle_2)
        if x1<R1*cos(Angle_2(j))+spx2
        F_spring(j)=-F_spring(j);
        end
```

```
end
% newv
for i=1:length(Angle_2)
if (R1^2+l_spring(i).^2-
R2.^2)/(2*R1*l_spring(i))<1
Fv_angle(i)=acos((R1^2+l_spring(i)^2-
R2.^2)./(2*R1*l_spring(i)));
end
if (R1^2+l_spring(i).^2-
R2.^2)/(2*R1*l_spring(i))>=1
    Fv_angle(i)=0;
end
end
Tan_angle1=.5*pi-Fv_angle;
M_hum=((F_spring+Voorspanning).*cos(Tan_ang
le1));
Tan_angle2=1/2*pi-(pi-abs(Angle_1-Angle_2)-
Fv_angle);
M_motor=((F_spring+Voorspanning).*cos(Tan_a
ngle2)*R2*.1)./((9.81));

for z=1:length(Angle_2)
     if l_spring(z)-l_spring_rust>max_l
         M_hum(z)=0;
         M_motor(z)=0;

     end
end
deltaAngle=abs((Angle_1)-(Angle_2));

Theta=acos((-
R2.^2+l_spring.^2+R1^2)./(2*l_spring*R1));
Thetat=1/2*pi-Theta;
Mhuman=M_hum*R1/Rhum;%(F_spring+Voorspannin
g).*cos(Thetat);%*R1;
[~,top]=max(Mhuman);
```

Function fit

```
 p1 = -2.1931e-06
  p2 = 0.00064332
  p3 = -0.065656
  p4 = 2.7741
  Angle_ratio=rad2deg(Angle_1)
 RatioAngle(g)= p1*Angle_ratio^3 +
p2*Angle_ratio^2 +p3*Angle_ratio + p4

p(:,g)=polyfit(((M_hum(1:top))*R1/Rhum).*Ra
tioAngle(g),rad2deg(Angle_2(1:top))*1000,2)
;
 p(1,:)=round(p(1,:),1);
 p(2,:)=round(p(2,:));
 p(3,:)=round(p(3,:));


q(:,g)=polyfit(((M_hum(1:top))*R1/Rhum).*Ra
tioAngle(g),Angle_2(1:top)*-
276.12+1111.1,2);
 q(1,:)=round(q(1,:),3);
 q(2,:)=round(q(2,:),2);
 q(3,:)=round(q(3,:));


end

  ArduinoRatioAngle=round(RatioAngle*1000)
F_moment_hum=((F_spring+Voorspanning).*cos(
Angle_tan));     %N*mm
F_moment_motor=((Voorspanning+F_spring).*co
s(Angle_tan))/9.81;   %N*mm
maxhuman(v,h)=max(M_hum)*R1/Rhum
maxmotor(v,h)=max(M_motor)%/20/Rendement

Pinchforcehelp(v,h)=((max(M_hum)*R1/Rhum)).
/(1.9018)*1.2204/1.7469%(2.1017)*1.2623/1.9
301%((cos(0.9194)+1.1998.*
0.9562))*(1.1998/1.7035)
r1length(v,:)=[R1-1.5]
r2length(:,h)=[R2+1.5]
end

end
```

Published with MATLAB® R2017b

Appendix H. Electric schemes

In Figure H.1 the PBC for the Arduino and motor driver has been shown. The green boxes (green line with black and green line with white) are connection sockets for the sensors, the motor and the power source. For the force sensor an amplifier is needed to read the signal, an INA125p amplifier has been used. In Figure H.2 the control scheme has been shown for this amplifier. To summarize in figure H.3 the complete electric scheme of the Hybrid Hand has been shown.

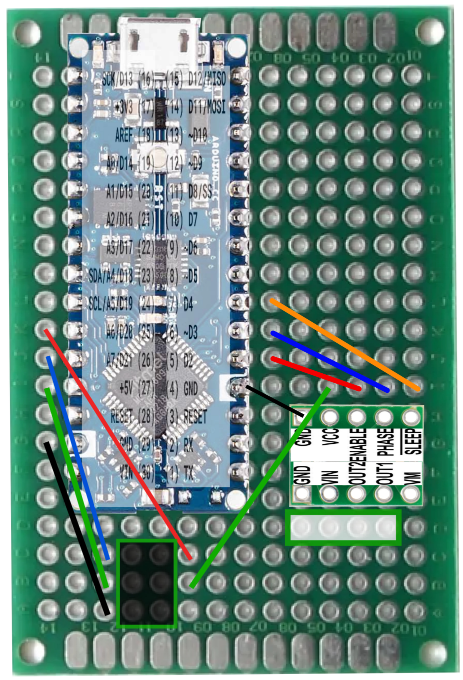


Figure H.1: PCB of the Arduino and motor driver.

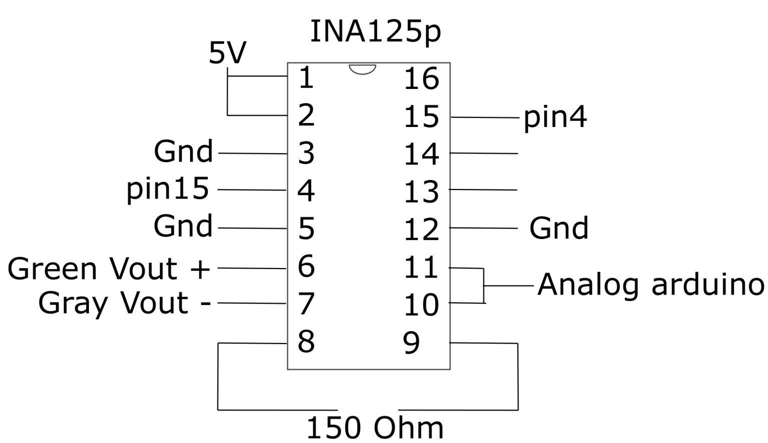


Figure H.2: Connection scheme of the INA125p amplifier

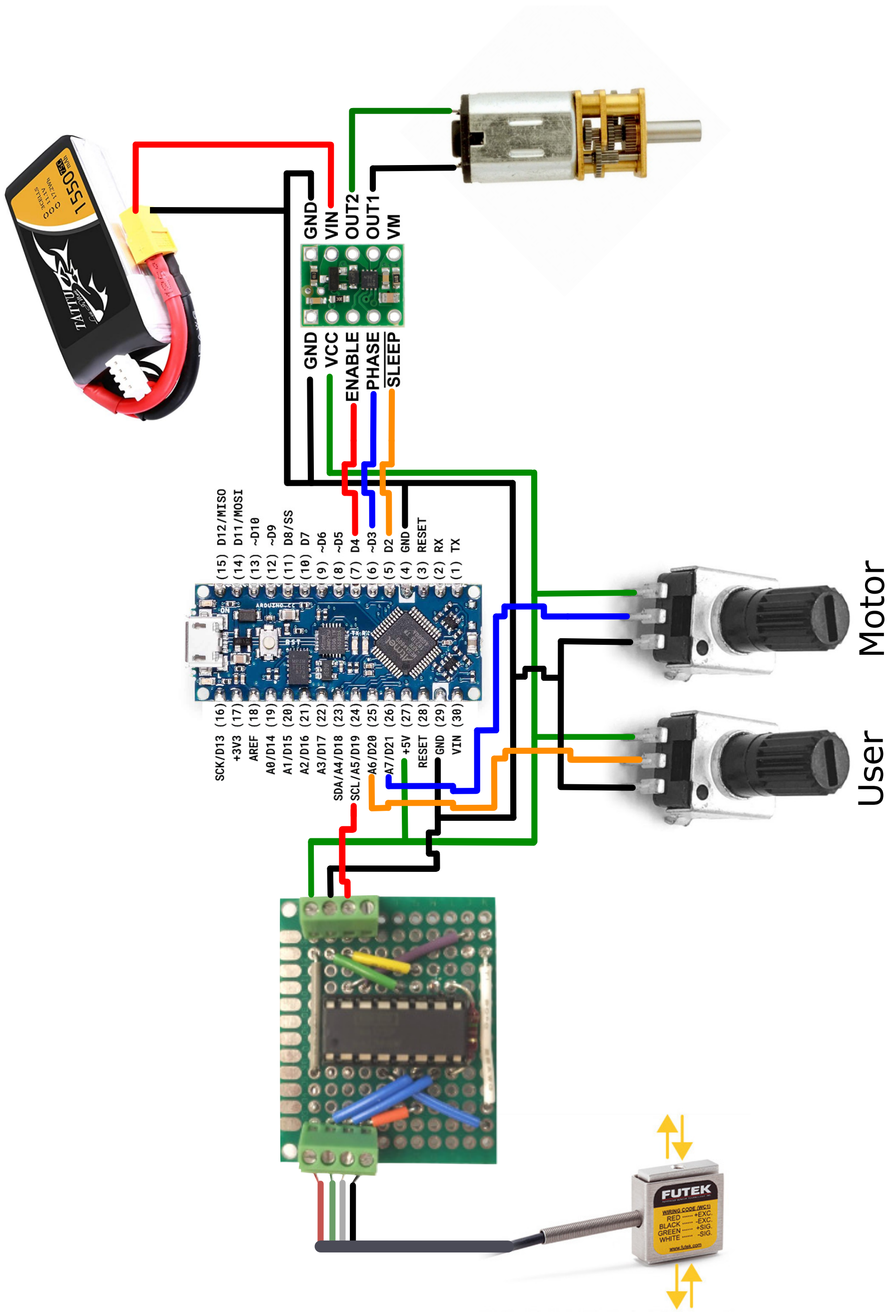


Figure H.3: Electric scheme of the Hybrid Hand with all the components and sensors included.

Appendix I. Static simulation logs

Logs of the static tests with Tinkercad virtual simulator. 4 Situations have been used in this simulator. After all of the results of the static tests were successful the actual prototype has been tested.

Test case	User Location	Motor Location	Forcesensor	Output
1 (successful)	800	800 (890 after 1^{st} loop)	0 N	Expected: 0N Force, Hand open to location 890 Sim: Forcesensor [N] =0 Force Determined Position user =800 Ratio Run=829 Det ratios Location user800 Force<3, motor run to:890 Motor open to:890 new location motor! =890
2(successful)	800	800	42(5 N)	Expected:5N Force, Hand passive closing (motor on to close hand) Sim: Forcesensor [N] =5 Force Determined Position user = 800 Position motor = 800 Ratio Run=0.63 Pinch_user = 3 Motor Amplification = 3 Det ratios Force 3-10 Passive closing
3(successful)	800	800 (734 after 1^{st} loop) (727 after 2^{st} loop)	125(15 N)	Expected: 15N force, location expected 727 Sim: Forcesensor [N] =15 Force Determined Position user = 800 Position motor = 800 Pos_user_deg=65 Ratio Run=0.62 Pinch_user = 9 Motor Amplification = 9 Force > 10 Motortarget determined = 727 PD value = 734 error =73 Motor closing to:734 Motor location =800 new location motor! =734 Forcesensor [N] =15 Force Determined Position user = 800 Position motor = 734 Index=64 Pos_user_deg=65 Ratio Run=0.62 Pinch_user = 9 Motor Amplification = 9 Force > 10 Motortarget determined = 727 A=0.07 Amplification = 9 PD value = 727 Motor closing to:727 Motor location =734 new location motor! =727

4(successful)	800	800
		(701 after 1st loop)
		(690 after 2nd loop)
			300(34 N)
			
			Max(pinch)
			= 15N
				Force: 34 N, 15N max amplification so expected
				Motor location 690
				
				`Forcesensor [N] =34`
				`Force Determined`
				`Position user = 800`
				`Position motor = 800`
				`Pos_user_deg=65`
				`Ratio Run=0.62`
				`Pinch_user = 21`
				`Motor Amplification = 21`
				`Det ratios`
				`>10`
				`NEW MOTOR AMP`
				`Motortarget determined = 691`
				`Amplification = 15`
				`PD value = 701`
				`kp = 0.90`
				`error =109`
				`Motor closing to:701`
				`new location motor! =701`
				`Motor Amplification true = 0`
				`300`
				`Forcesensor [N] =34`
				`Force Determined`
				`Position user = 800`
				`Position motor = 701`
				`Index=64`
				`Pos_user_deg=65`
				`Ratio Run=0.62`
				`Pinch_user = 21`
				`Motor Amplification = 21`
				`Det ratios`
				`>10`
				`NEW MOTOR AMP`
				`Motortarget determined = 691`
				`Amplification = 15`
				`PD value = 692`
				`kp = 0.90`
				`error =10`
				`Motor closing to:692`
				`new location motor! =691`

Appendix J. Arduino script control of the Hybrid Hand

The control system of the Hybrid Hand is shown below. This Arduino script is used for the amplification tests and the functionality of the hand.

```
// Initialisation Variables
volatile unsigned short int
Pos_user = 0;       //Position
user
volatile unsigned short int
Pos_motor = 0;      //Position
Motor
volatile short int error =
0;          //error for
Pcontroller
volatile byte Index_array =
0;    //Index of the formula array
volatile byte Pos_user_deg=0;
//position user in degree
volatile float
RatioFinFout=0;  //Ratio
volatile byte Pos_motor_deg=0;
//position motor in degree
volatile unsigned short int
Forcesensor=0;//Forcesensor value
volatile byte
Force=0;           //Actual Force
volatile short int
Amplification=0; //Amplified
target force
volatile byte
Motor_Amplification=0; //Motor
amplification
volatile short int
Motor_Amplification_true=0;//Actua
l motor amplification
volatile short int Pinch_user=0;
//user pinch force
volatile int Target_motor_pos=0;
//target motor position
volatile byte mode=0;
bool passive=false;
//volatile float val_a=0;
//volatile float val_b=0;
//volatile float val_c=0;
//positional constants
const short int Open_pos_user =
199;
const short int Closed_pos_user =
500;
const short int Open_pos_motor =
860; //Position of the motor to
open the hand
const short int Closed_pos_motor =
470; // position of the motor to
close the hand
//Constants
//Polynomial constants and Ratio
vectors
//I(0) = 129 degrees, I(84) =45
degrees
// A, B, C are TF Motorlocation
Degrees=(Ax^2+bx+c)/1000 x=Wanted
pinchforce
//Ratio = Fin[mN]/Fout[N]
// A*1000, B*-10, C=C, Ratio*1000
const byte Number_I = 85;
const int
A[Number_I]={304,287,272,258,246,2
36,226,218,210,203,197,191,186,181
,177,173,169,166,162,159,157,154,1
52,149,147,145,143,141,140,138,136
,134,133,131,130,128,127,125,123,1
22,120,119,117,115,113,112,110,108
,106,104,102,100,98,96,94,92,90,87
,85,83,81,78,76,74,72,69,67,65,63,
61,58,56,54,52,50,48,46,44,42,41,3
9,37,36,34,33};
const short int
B[Number_I]={158,153,149,145,142,1
39,136,133,131,129,127,125,123,122
,120,119,118,116,115,114,113,112,1
11,110,110,109,108,107,107,106,105
,105,104,104,103,102,102,101,100,1
00,99,98,98,97,96,96,95,94,93,92,9
1,90,90,89,88,87,86,85,83,82,81,80
,79,78,77,75,74,73,72,70,69,68,67,
65,64,63,61,60,59,58,56,55,54,53,5
2};
const short int
C[Number_I]={482,487,492,496,501,5
06,511,516,521,525,530,535,540,545
,549,554,559,564,569,574,578,583,5
88,593,598,602,607,612,617,622,627
,631,636,641,646,651,655,660,665,6
70,675,680,684,689,694,699,704,709
,713,718,723,728,733,737,742,747,7
52,757,762,766,771,776,781,786,790
,795,800,805,810,815,819,824,829,8
34,839,843,848,853,858,863,868,872
,877,882,887};
const short int
Ratio[Number_I]={302,311,320,328,3
36,343,350,357,363,370,375,381,386
,391,396,401,405,409,413,417,421,4
24,428,431,434,437,440,443,446,449
,451,454,457,460,462,465,468,471,4
74,477,480,484,487,491,494,498,502
,507,511,516,521,526,532,537,543,5
50,556,563,571,578,586,595,603,613
,622,632,643,654,665,677,689,702,7
16,730,744,759,775,791,808,825,844
,862,882,902,922};


//Pd-controller
const byte Factor_A = 15;
//Amplification factor *10
const byte kp = 5; //Prop constant
*10
volatile long Motor_target_PD=0;
float Tussenvalue=0;
volatile short int
Amplification_PD=0;
const byte Motor_power = 255;
```

```
const byte Max_Pos_error = 10; //
max error for position [
const byte Max_F_error = 2; // max
error for force [N]
const byte Max_motor_amp = 20; //
Max motor Amplification
const byte Max_Pos_error_open =
10;


  // Pin constants
  const byte Motor_Phase= 3;
//Motor Phase pin
(forward/backwards)
  const byte Motor_Enable= 2;
//Motor Enabled
  const byte Motor_Speed=4;
//Motor speed input pin (Sleep)
  const short int Pin_potmotor=
A6;
  const short int Pin_potuser=A7;
  const short int Pin_Force=A3;

////Simulation values
//volatile int analogmotor = 800;
//const int analoghuman = 800;
//const int analogForce = 300;
//bool test=false;

void setup() {
Serial.begin(9600); // seriele
poort aan op 9600 baud.
pinMode(Motor_Phase, OUTPUT);
//Phase output
pinMode(Motor_Enable, OUTPUT);
//Enable output
pinMode(Motor_Speed,
OUTPUT);//Sleep output
// motor disabled
digitalWrite(Motor_Enable, LOW);
//Motor disabled
digitalWrite(Motor_Speed, LOW); //
Motor disabled

Serial.println("Initializing...");
while (analogRead(Pin_Force)>200){
Serial.println("Don't apply force
please");
delay(100000);
}
while
(analogRead(Pin_potuser)<Open_pos_
user-20 ||
analogRead(Pin_potuser)>Closed_pos
_user+20){
Serial.println("Human sensor
error");
Serial.println(analogRead(Pin_potu
ser));
delay(100000);
}
while
(analogRead(Pin_potmotor)<Closed_p
os_motor-20 ||
analogRead(Pin_potmotor)>Open_pos_
motor+20){
Serial.println("Motor sensor
error");
Serial.println(analogRead(Pin_potm
otor));
Motor_run(500);
delay(100000);
}

}

//Function to determine Ratio user
float Ratio_run(short int
Pos_user){
  //Function
  Pos_user_deg=round(Pos_user*.207
42+17.7065);
  Index_array=129-Pos_user_deg;
  RatioFinFout=Ratio[Index_array]*
0.001;
  Serial.print("Index=");
  Serial.println(Index_array);
  Serial.print("Pos_user_deg=");
  Serial.println(Pos_user_deg);
  Serial.print("Ratio Run=");
  Serial.println(RatioFinFout);



  //delay(10);
  return RatioFinFout;

}



//Function to find motorlocation
0-1023 for target amplification
int Motortarget_run(short int
Motor_Amplification, short int
Pos_user){
  //Motor location in 0-1023
 // val_a=A[Index_array]*0.001;
 // val_b=-B[Index_array]*.1;
 // val_c=C[Index_array];
  //Serial.print("Array values");
  //Serial.println(val_a);
  //Serial.println(val_b);
  //Serial.println(val_c);
  error=Motor_Amplification-
Amplification_PD;
  Amplification_PD=Amplification_P
D+kp*error*.1;
  Tussenvalue=(0.001*A[Index_array
]);
```

```
  Target_motor_pos=
int(round((Tussenvalue*Amplificati
on_PD*Amplification_PD)+(Amplifica
tion_PD*-
B[Index_array]*.1)+C[Index_array])
);
  Serial.print("PD value [N] = ");
      Serial.println(Amplification
_PD);
      Serial.print("error =");
      Serial.println(error);
  Serial.print("Motortarget
determined = ");
  Serial.println(Target_motor_pos)
;
  // Serial.print("A=");
  //Serial.println(A[129-
Pos_user_deg]*0.001);
  //delay(10);
  return Target_motor_pos;
  }



int Forcesensor_run(short int
Forcesensor){
  Force=int(round(0.11*
Forcesensor+0.76));

  //delay(10);
  if (Forcesensor<19){
  Force=0;
  }
  Serial.print("Forcesensor [N]
=");
  Serial.println(Force+Motor_Ampli
fication_true);
  return Force;
}



// Motor function position
void Motor_run(short int
valTarget){
  // Move motor Open
  while (Pos_motor <=
valTarget  &&
abs(analogRead(Pin_potmotor) -
valTarget) > Max_Pos_error_open){
//lower than target
      Pos_motor=(analogRead(Pin_po
tmotor));// - Min_Pot)*
1023/(Max_Pot - Min_Pot) ;
      if
(Pos_motor>Open_pos_motor){
        analogWrite(Motor_Speed,
0);
        Serial.print("error1");
        break;
      }
      if (Pos_motor>valTarget){
        break;
      }
      Forcesensor=Forcesensor_run(
analogRead(Pin_Force))+Motor_Ampli
fication_true;
       if((Forcesensor-
Motor_Amplification_true>3) &&
(mode==1)){
       Serial.print("Mode =");
       Serial.println(mode);
        mode=0;
        analogWrite(Motor_Speed,
0);
        Serial.print("error2");
        break;
      }
      error =  abs(valTarget -
analogRead(Pin_potmotor));
      //P-controller
      //MotorPWM = ((kp *
error))/100 + Min_power;
      analogWrite(Motor_Speed,
255);
      analogWrite(Motor_Enable,
255);
      digitalWrite(Motor_Phase,
LOW);
      //int sensorValue =
analogRead(Pin_Force); // Lees de
analoge ingang uit.
      //int forcevalue =
sensorValue*0.11+0.83;
      Serial.print("Motor open
to:");
      Serial.println(valTarget);
      Serial.print("Current
location");
      Serial.println(analogRead(Pi
n_potmotor));

  }


//Move motor closed
  while (Pos_motor >= valTarget &&
abs(analogRead(Pin_potmotor) -
valTarget) > Max_Pos_error){
      Pos_motor=(analogRead(Pin_po
tmotor));// - Min_Pot)*
1023/(Max_Pot - Min_Pot) ;
      error =  abs(valTarget -
Pos_motor);
      Forcesensor=Forcesensor_run(
analogRead(Pin_Force))+Motor_Ampli
fication_true;
      if((Forcesensor-
Motor_Amplification_true<3) ||
```

```
(Forcesensor-
Motor_Amplification_true>10) &&
(mode==2)){
        mode=0;
        break;
      }
      if((Forcesensor <10) &&
(mode==3)){
        mode=0;
        break;
      }

      //P-controller
      //MotorPWM = ((kp *
error))/100 + Min_power;
      analogWrite(Motor_Speed,
255);
      analogWrite(Motor_Enable,
255);
      digitalWrite(Motor_Phase,
HIGH);
    Serial.print("Motor closing
to:");
      Serial.println(valTarget);
      Serial.print("Motor location
=");
      Serial.println(Pos_motor);

  }
analogWrite(Motor_Speed,
0);  //turn motor off

//delay(10);
}

//void Pos_calculation(short int
Forcesensor){
  //Pos_user = (analoghuman);
//Read Position user
  //Pos
  //Index_array = floor((Pos_user-
Closed_pos_user)/( (Open_pos_user-
Closed_pos_user)/Number_I));
//calculate the fit index for the
abc array
  //float
target_force=Forcesensor+Versterki
ngsfactor;

  //Solve abc-formula
  //float
discriminant=sqrt(B[Index_array]-
4*A[Index_array]*(C[Index_array]-
target_force));
  //float target_position=(-
B[Index_array]+discriminant)/2*A[I
ndex_array];



//}
```

```
//
void loop() {
  //Determine force and user
location
  Serial.print("forcesensor =");
  Serial.println(analogRead(Pin_Fo
rce));
  Forcesensor=Forcesensor_run(anal
ogRead(Pin_Force))+Motor_Amplifica
tion_true;
  Serial.println("Force
Determined");
  Pos_user=analogRead(Pin_potuser)
;
  Pos_motor=analogRead(Pin_potmoto
r);
    Serial.print("Position user =
");
    Serial.println(Pos_user);
    Serial.print("Position motor =
");
    Serial.println(Pos_motor);
  //delay(10);
  //Determine ratios and
amplification
  RatioFinFout=Ratio_run(Pos_user)
;
  Pinch_user=round(Forcesensor*Rat
ioFinFout);
    Serial.print("Pinch_user = ");
    Serial.println(Pinch_user);
  //Serial.print("MULTIPLICATION")
;
  //Serial.print(Forcesensor);
  //Serial.println(RatioFinFout);
  Amplification=int(round(Pinch_us
er*Factor_A*.1));
  Motor_Amplification=Amplificatio
n-Pinch_user;
    Serial.print("Motor
Amplification = ");
    Serial.println(Motor_Amplifica
tion);

    Serial.print("Motor
Amplification true = ");
    Serial.println(Motor_Amplifica
tion_true);
    //delay(10);

  //Safety that motor
amplification not higher than max
motor force
  if
(Motor_Amplification>Max_motor_amp
) {
    Motor_Amplification =
Max_motor_amp;
    }
  //Serial.print("Location user");
  //Serial.println(analoghuman);
```

```
 //delay(10);

if (Forcesensor-
Motor_Amplification_true<=3){
    mode=1;
    Motor_Amplification_true=0;
    Serial.print("<3, motor run
to:");
    Serial.println(Open_pos_motor)
;
    Forcesensor=Forcesensor_run(an
alogRead(Pin_Force));
    //delay(10);
   if (abs(Pos_motor-
Open_pos_motor)>Max_Pos_error){

   Motor_run(-
Pos_user+1070);  //avg=1026  1060
okay
   }
   Motor_Amplification_true=0;
  }
//Serial.print("Check");
//Serial.print(Forcesensor-
Motor_Amplification_true>3);
//Serial.print(Forcesensor-
Motor_Amplification_true<10);
//Serial.println(Pos_motor>=Closed
_pos_motor);
if ((Forcesensor-
Motor_Amplification_true>3) &&
(Forcesensor-
Motor_Amplification_true<10)){
  Serial.println("Force = 3-10
N");
}
while ((Forcesensor-
Motor_Amplification_true>3) &&
(Forcesensor-
Motor_Amplification_true<10) &&
(Pos_motor>=Closed_pos_motor)){
      mode=2;
      Serial.println("3-10");
      Serial.println("Passive
closing");
    Motor_run(-
Pos_user+1026);            //avg=1
026
    Pos_motor=(analogRead(Pin_potm
otor));
    Forcesensor=Forcesensor_run(an
alogRead(Pin_Force));
    Motor_Amplification_true=0;
    passive=true;
}
if ((Forcesensor<=3) &&
(passive=true)){
    delay (100);
    passive=false;
}
if ((Forcesensor-
Motor_Amplification_true >=10)){
    Serial.println(">10");
  mode=3;
  Pinch_user=round(Forcesensor*Rat
ioFinFout);
  Amplification=Pinch_user*Factor_
A*.1;
  Motor_Amplification=Amplificatio
n-Pinch_user;
  //Safety that motor
amplification not higher than max
motor force
  if
(Motor_Amplification>Max_motor_amp
) {
    Motor_Amplification =
Max_motor_amp;
    Serial.println("NEW MOTOR
AMP");
    }
  Target_motor_pos=Motortarget_run
(Motor_Amplification,Pos_user);
      Serial.println("Amplificatio
n = ");
      Serial.println(Motor_Amplifi
cation);


  Motor_run(Target_motor_pos);
  Motor_Amplification_true=round(A
mplification_PD/RatioFinFout);
      Serial.print("Motor
Amplification true = ");
      Serial.println(Motor_Amplifi
cation_true);
  passive= false;
}

    //(Motor_Speed, 0);  //turn
motor off



}
```