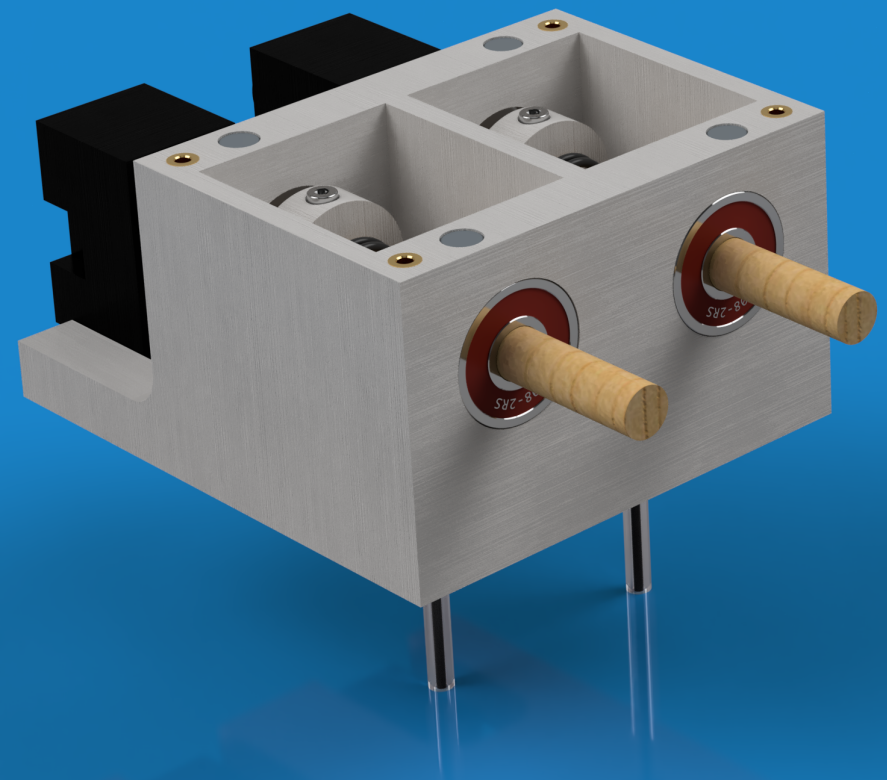
Technical University of Denmark

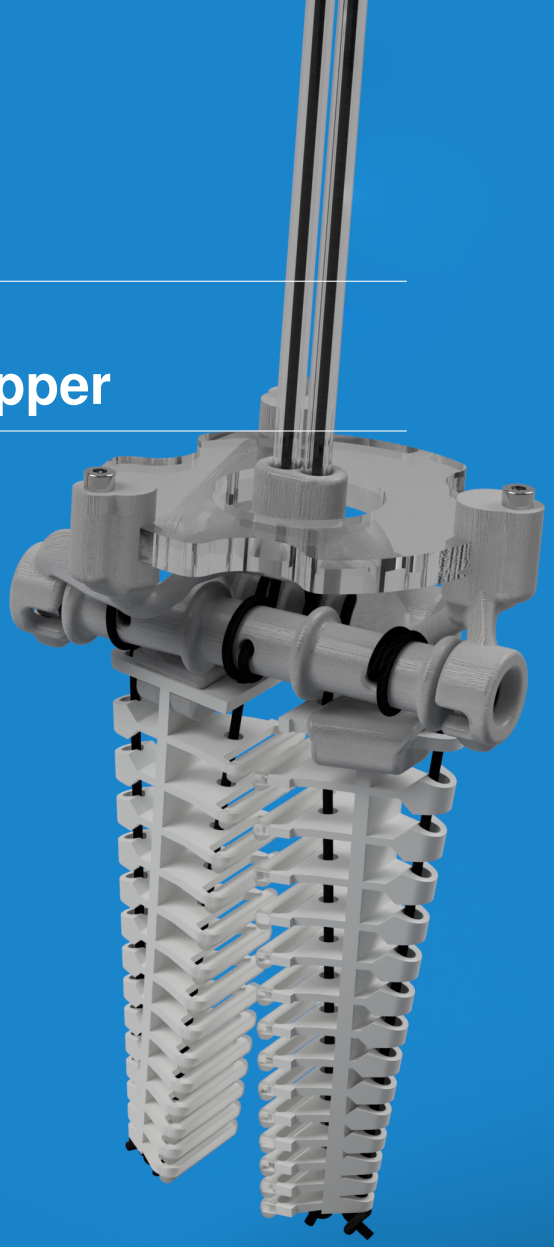
Design and development of a Bio-Inspired, Lightweight Soft Robotic Gripper

BACHELOR THESIS

Mads August Claussen
s223952

Emil Broge Johansen
s224025





January 18, 2026

Abstract

Soft robotics offers inherent compliance and adaptability, enabling safe interaction with fragile and irregularly shaped objects. However, this compliance often comes at the cost of structural integrity and low payload capacity, a limitation that is also present in soft continuum robot arms. Because of the low payload capacity of such soft robot arms, mounted end-effector are constrained by weight and space limits. Therefore there is a need for the development of a lightweight soft robotic gripper.

This report presents the design and prototyping of a lightweight, servo-motor-driven, tendon-actuated soft robotic gripper. The gripper consists of two silicone coated monolithic 3D-printed TPU fingers. The fingers are mounted on a 3D-printed PLA finger base incorporating miniature tendon drums and bearings, enabling compact tendon force distribution without the bulk of normally used tendon distribution mechanisms. Actuation and control are handled by an externally mounted module consisting of a microcontroller and servo motors, employing a bio-inspired control strategy based on octopus-like wrapping motion.

The gripper is experimentally evaluated using payload and grip success tests on a defined set of test objects. The final gripper weighs 37 grams and achieves a maximum lifting capacity of approximately 100 grams in pinch grasping and up to 750 grams in wrapping grasping. In figure 1 the main components of the system are presented graphically.

The code used, the STL files for 3D print and the raw test data can be found in the public GitHub repository: `https://github.com/emilbroge/lightweight-soft-robotic-gripper-bsc-thesis`. Code has also been uploaded with the report as External Appendix I.

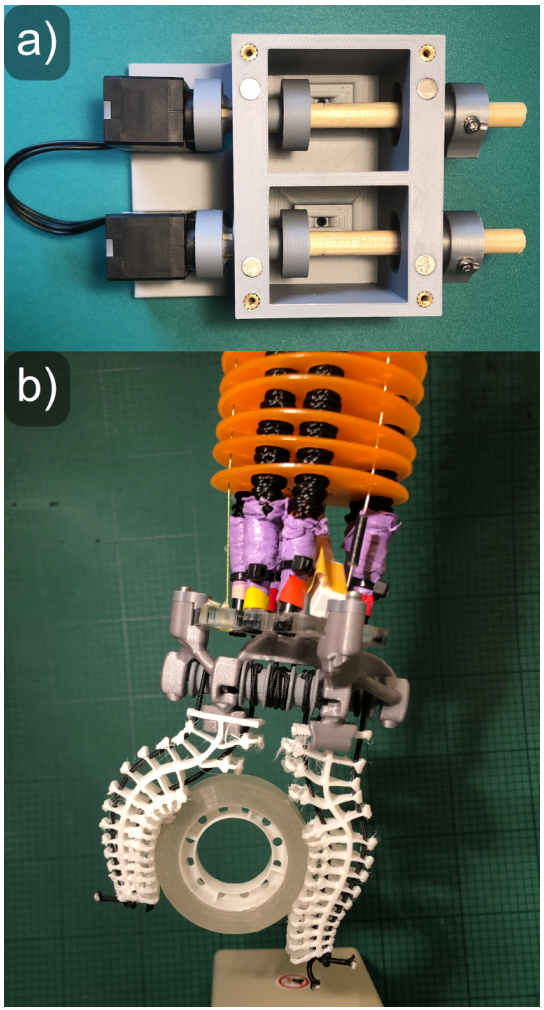
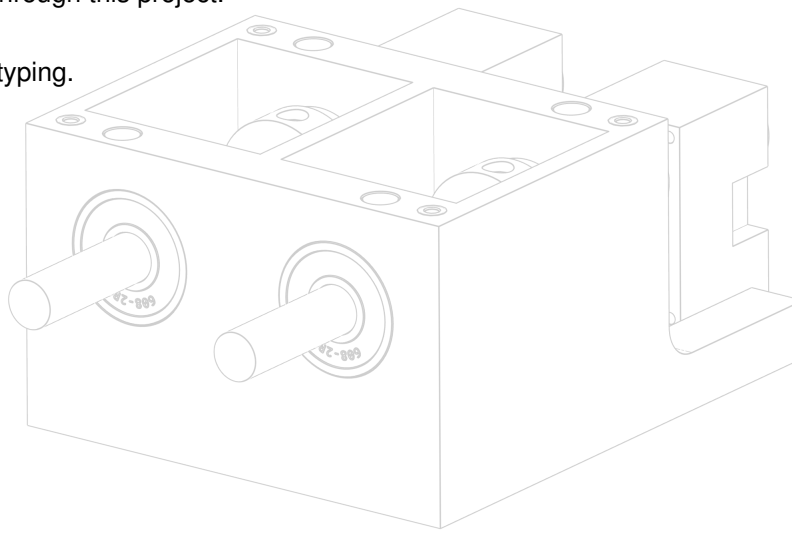


Figure 1: **a)** Actuator module, mounted external to the arm **b)** Gripper mounted on the NRT Lab arm - actuator and gripper are connected via two tendons running in PTFE tubes

Acknowledgements

We would like to thank our supervisors Silvia Tolu, Torben Lenau and Søren Hansen for guiding us through this project.

We would like to thank our friend Jakob Guarini for letting us use his 3D printer extensively for prototyping.



Part I

Project

Table of Contents

1 Introduction

Because of its inherent compliance and ability to handle complex objects, soft robotics is an emerging field within robotics. Soft robotics focuses on the use of deformable and flexible materials in place of conventional rigid metals and plastics. Beyond material choices, soft robotics often uses alternative actuation methods such as flexible fluid actuators (FFA) or tendon-driven mechanisms, rather than traditional rigid gears and links mechanisms. In the case of grippers, soft robotic designs enable solutions that inherently adapt to objects of varying shapes, sizes, and friabilities while maintaining safety in human environments. However, the inherent compliance of soft robots also limits their strength and load-bearing capacity, mainly due to a lack of mechanical rigidity. Therefore, it is desirable to develop a lightweight, soft robotic end-effector that can be mounted onto low-strength, soft robotic arms.

1.1 Problem statement

To enable the gripping of diverse objects, the gripping mechanism needs to conform properly to the gripped objects. In traditional rigid robotics, this is often achieved using many sensors, actuators and complex control techniques. The use of biologically inspired design realized through compliant materials and mechanisms is advantageous for performing this gripping task. The use of a compliant mechanical design should also reduce the amount of complex control algorithms needed. However, exactly because of this compliance (lack of rigidity), the mechanism is likely to be less predictable which can pose a potential challenge for precise movements. The project aims to investigate:

- How has a similar gripping task been solved previously in literature?
- Which mechanical mechanisms and materials are suitable for this task?
- What electrical framework is needed to realize such a system?
- How can the performance of this gripper be evaluated?

2 Project overview & Design Requirements

2.1 Project scope and goals

This project aims to develop a novel end-effector for the Neuro Robotics Technology Lab (NRT-Lab) continuum robot arm. Currently, the robot arm uses velcro tape as the interface between arm and payloads. The robot arm has previously been tuned and controlled with payloads up to maximum 100 grams. This means that the developed gripper should weigh as little as possible, if any payload at all is to be lifted.

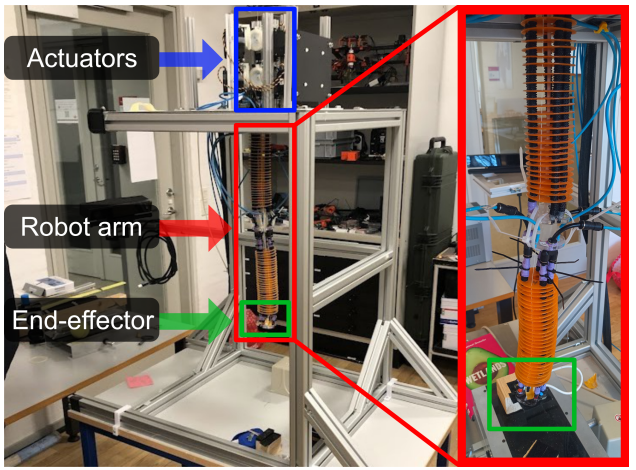


Figure 2: The soft continuum robot arm in the Neuro Robotics Technology Lab (NRT-Lab). This is the robot arm that we need to develop an end-effector for.

The NRT arm is attached to a metal "enclosure", where the motors moving the arm itself are displaced. This means that we in the design of the end-effector *can* utilize this space - if needed. Eg. a displacement of the actuators, to save weight on the end-effector itself.

The project aims to accomplish the following high-level goals:

- End up with a lightweight first prototype of the end-effector
- Grip and hold certain test objects (to be defined)
- Actuate the gripper automatically and fast
- Test the gripper's functionality when mounted on the NRT robot arm

2.2 Project methods

The project employs a systematic approach to developing the gripper, beginning with research into existing robotic grippers and biological systems for inspiration. After establishing an informed initial design direction, the development follows an iterative process, which also dictates the structure of this report:

1. **Conceptual design**: Draw on academic literature, bio mimicry, and structured brainstorming to generate concepts.
2. **Embodied design**: Realize selected concepts as prototype(s) using CAD, 3D printing, silicone casting, and related fabrication techniques.
3. **Testing & results**: Perform experimental testing to quantify performance (e.g., grasp success, payload) and compare results against project targets.

While the electrical, control and mechanical design processes are intertwined, the above mentioned process is especially important to the mechanical design process.

What is problem-driven, bio-inspired design?

Problem-driven, bio inspired design (BID) is a methodology to make innovative engineering solutions, by looking to nature for inspiration. This approach has been shown to significantly increase the generation of highly novel concepts compared to traditional design methods [1]. There are five phases in the framework: [2]

1. **Problem Analysis:** The technical problem is defined and the demands and criteria are abstracted by re-framing requirements (e.g., focusing on "attachment" rather than "glue").
2. **Search:** Potential biological strategies are identified through databases or literature, looking for organisms that have evolved to solve the abstracted functions.
3. **Understand:** The biological strategies are analyzed to understand the underlying principles.
4. **Transfer:** This is the bridge where the biological strategy is translated into technical principles, stripping away the biological context to focus on what is technically feasible.
5. **Design:** The transferred principles are implemented into a manufacturable engineering design and validated against the original technical needs.

We went through the first three steps in our preliminary work (External Appendix II), but when further developing the different subsystems of the gripper we will use this methodology as a framework to find the means to perform the desired functions.

Another reason to use bio inspired design, is making communication easier, since it is more tangible to relate to something that resembles nature. This also makes naming easier and more systematic.

2.3 Prior work

This thesis builds directly upon the preliminary research that we conducted in the special course report "Soft Robotics Gripper Review" (see External Appendix II). The report investigates a wide variety of grippers found in literature and nature. This includes analyzing the different technologies and methods used in different types of grippers.

The prior work established a framework for analyzing soft grippers by defining five subsystems, enabling us to gain an overview of the design space. The five subsystems that we identified were:

1. **Dry adhesion:** This includes different technologies to better adhere to a surface, while being able to remove the adhesion quickly again. The

main dry adhesion methods found were silicone, electro adhesion and gecko-skin.

2. **Gripper Morphology:** This is the geometry, mechanical layout and kinematics of the gripper. Grippers in nature were described using its degrees of freedom, contact conditions and geometry.
3. **Actuation:** This is the means to which a control signal is converted to movement. We looked into DC motors, pneumatics, shape memory alloys and dielectric elastomers.
4. **Control:** The strategies used to control the gripper. This mainly involves the properties of the control signal to the actuators.
5. **Energy Supply:** The power infrastructure needed to control the gripper. Could be simple low voltage batteries, high voltage converts or pressure reservoirs.

These different categories are of course intertwined, but listing them like this, made it easier to navigate in the vast amount of soft grippers.

Evaluated Design Concepts

The previous study transformed these technologies into two distinct design concepts, serving as the departure point for this project:

- **Compliant Tendon Design:** This concept mimics the human hand by using DC motors to pull artificial tendons through compliant, underactuated fingers. It was found to offer high controllability and sensor feedback (via motor current/encoders) but introduced mechanical complexity regarding friction and tendon routing.
- **Pneumatic Design:** This concept utilizes a silicone elastomer structure, fabricated via 3D-printed molds, which is actuated by a remote external pressure reservoir and controller (to minimize end-effector mass). Compliance and movement are managed through internal geometry or particle jamming. This offers a scale of complexity from a single-DOF system to multi-compartment fingers for independent control. Because pneumatic systems lack inherent feedback, integrated sensors (e.g. piezo-resistive sensors) are required to monitor bending angles.

While dry adhesion, such as gecko skin or electro adhesion, can be integrated to enhance the contact conditions for both concepts, these additions introduce fabrication complexity. For gecko skin we have directional release constraints, and for electro adhesion high-voltage safety concerns. Therefore, we chose not to go forward with a focus on dry adhesion.

The conclusion of the previous report demonstrated that no single subsystem offers a universal solution; rather, the design must balance fabrication complexity with grip strength and weight.

Choosing design concept from the prior work

To focus the scope of this project, a specific design concept was selected from the preliminary research. To do this while mitigating subjective bias, we used a weighted decision matrix to quantify the performance of each design concept (appendix H). To test the reliability of our results, we introduced two alternative weighting sets: one emphasizing cost and the other focusing on strength and efficiency. These variations acted as a sensitivity check for our decision. The results in Table 1 show that the tendon-driven concept consistently outperformed the alternatives, leading us to proceed with this design.

Score type	Tendons	Pneumatics
Primary weighted score	4.21	4.01
Weighted score with strength sensitivity	4.35	3.92
Weighted score with cost sensitivity	4.24	4.09

Table 1: The primary and weighted scores from appendix H for tendon and pneumatic systems. We made one score heavily emphasizing strength, and one including cost. The scale goes from 1 to 5.

2.4 Requirements

To make the design and evaluation process easier, we defined some specific requirements, stemming from the problem definition. This serves as a guideline when taking design decisions. We will go through these now. They are also formulated in table 2.

Requirement	Measure	Justification
Gripper weight	max. 40 g	Not to overcome the max payload of the robot arm
Load bearing	up to 60 g	To utilize the whole weight from 40 g (gripper weight) up to 100 g (max. payload)
Reliably grasp and hold test objects	90% success rate	-
Handle both rigid and soft objects	True/False	To grasp all the test objects
Grasp width	0.5-3 cm	To grasp all the test objects. Max width of test objects is 3 cm
Withstands repeated grasping of test objects without failure	-	To operate for longer, the gripper needs to be durable
Speed	< 3 s	For human interaction, measured from open state to closed state
Mountable on the end of the NRT Lab robot arm	True/False	This is the main task of the project
Transmission able to fit inside the NRT Lab robot arm	Diameter < 10mm	Transmission should be mounted internally, for a less intervening mounting.
Low Control Complexity	-	In interest of time and scope of the project. Only feed forward control, unless closed-loop is already implemented in motors.
Low Assembly Complexity	-	Since we have limited tools and equipment available, and for easy reproducibility. E.g. 3D printable.

Table 2: Overview of requirements of the gripper

As mentioned in the intro, the NRT-Lab robot arm can only support up to 100 grams including end-effector and payload. Because of this, we set a restriction of the weight of the gripper to 40 grams. This results in a required gripper load bearing of up to 60 grams. It is advantageous if the load bearing exceeds the 60 grams, since the gripper then potentially can be used on other robot arms.

As will be mentioned in the next subsection, we define some test objects to evaluate the gripper against. These test objects vary in size, shape and hardness. Thus the gripper needs to be able to handle these very different objects reliably.

As the NRT Lab robot arm is (also) designed for human interaction, the operation speed needs to be relatively fast. We assessed that a closing time below three seconds is sufficient as a novel target. Additionally, as the gripper needs to be mounted on the robot arm, this is obviously also a requirements - this includes being able to mount the transmission inside the arm and the gripper on the end of the arm.

As we have limited time and resources, the design and implementation complexity should be relatively low, this includes the control complexity and as-

sembly complexity.

Test Objects

We want to have a set of objects to test the gripper against. These objects should have varying parameters, such as mass, center of gravity, surface and shape, thus illustrating different challenges of gripping. Furthermore, they should be normal everyday items, as these pose a more realistic gripping challenge. As mentioned earlier, the maximum payload of the NRT-Lab arm is around a 100 grams. This means that the mass of the test objects should be at or below ca. 50-60 grams, since the max gripper weight is set to maximum 40 grams. In the following list we see a definition of the test objects. A graphical overview can also be seen in fig. 4.

- **Lego brick:** (3 g) This is a small item, requiring dexterity.
- **Sponge:** (5 g) A lightweight "squishy" object.
- **Tape:** (17 g) Allows wrapping through its center hole, but also poses a challenge if wrapped around, because of the low friction of the tape surface.
- **Cutlery:** (26 g) This is a difficult task since the cutlery is very thin, and the gripper has to resist a rotational moment.
- **Cloth**: (39 g) This object is very deformable.
- **Pill container:** (50 g) Object with flat surface and relatively heavy
- **Egg:** (55 g) Eggs are fragile (with unevenly distributed force) and therefore difficult to grasp without destroying if using a traditional gripper. Also, the mass of an egg (55 grams) is close to the maximum carrying capacity.

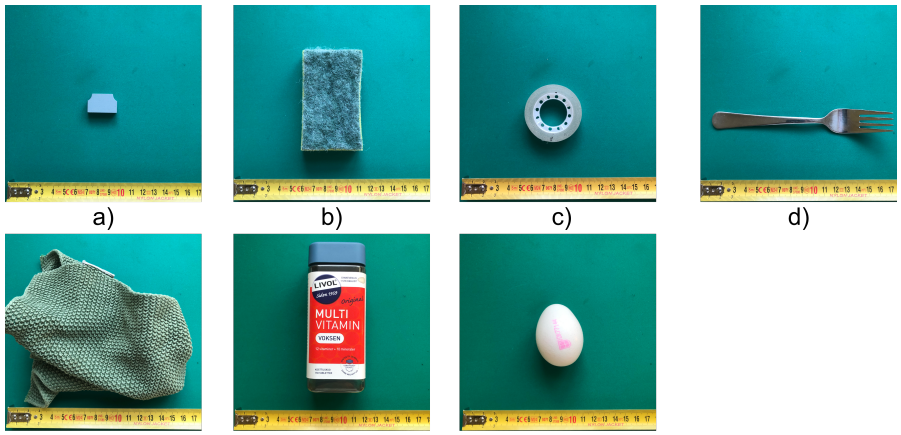


Figure 4: Test objects: **a)** "Lego" brick (3 g) **b)** Sponge (5 g) **c)** Tape (17 g) **d)** Cutlery (26 g) **e)** Cloth (39 g) **f)** Pill container (50 g) **g)** Egg (55 g)

Apart from these very realistic everyday test objects, we will in the testing of the gripper introduce a set of more *standardized* test objects. Here *standardized* refers to the fact that the object are simple shapes (box/sphere), with varying dimensions that we 3D print out of the same material every time. These objects will serve as a more quantitative test object.

This concludes the project overview and design requirements section. This is the basis for our further research of the problem and design space.

3 Previous Grippers

As a starting point for the design and development of the gripper, we take a look at how similar tasks have been solved previously. The prior work focused on general soft robotic grippers. Here we will limit ourselves to tendon driven robotic grippers. The amount of soft robotic grippers is large, but in the following we have selected some, that we feel are interesting specifically for our problem. In our research we found that the following topics are among the most relevant in (DC motor, tendon driven) soft robotic grippers:

- **Bio mimicry**: How are soft robotic grippers inspired by biology?
- **Balancing softness and rigidity**: Compliance and strength
- **Under actuation**: To what degree is underactuation used.
- **Amount of control**: How are the tendons actuated, sensing, modeling
- **Weight/size** of gripper, is actuator internal or external.

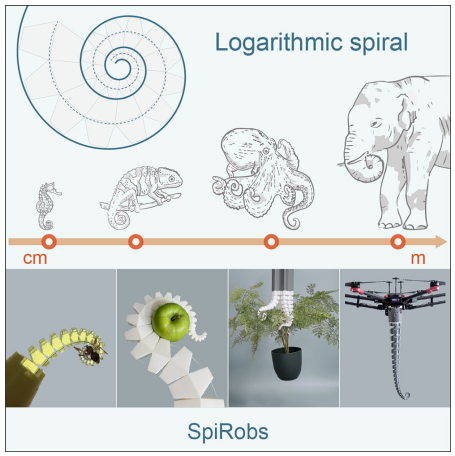


Figure 5: Screenshot of the graphical abstract of SpiRobs [3].

Wang et al. [3] created SpiRobs, a bio-inspired end-effector. It is based on the logarithmic spiral an octopus or elephant makes when curling their tentacle/trunk, and is able to grab, release and move, using only two tendons. A clever antagonistic multi-stage actuation strategy enables various different movement patterns, like reaching, curling up, grasping etc. The octopus design was both realized as a single-tentacle, combined arm + end-effector design, and as a multi-tentacle pure end-effector. The tentacles were realized as 3D printed thermoplastic polyurethane (TPU) with living hinges as the joints. The graphical abstract for SpiRobs can be seen in figure 5.

The team behind SpiRobs also made a follow-up article showing how they exploit friction to make different movement patterns. They made a mathematical model of the gripper in which they modeled the friction in the tendons inside the robot, using the Capstan equation ($\frac{F_{in}}{F_{out}} = e^{\mu\theta}$). By pulling with different amounts of force in the cables, they apply different amounts of tension inside the robot, and thereby "stiffness". Depending on the tension distribution inside the robot, different movement patterns can be created by pulling the different cables [4].

Crooks et al. [5] made a very simple two finger gripper using compliant mechanisms. It is inspired by the structure of fish fins (Fin Ray®). The gripper is made of TPU and actuated by a DC motor. The actuation happens through the compliant mechanism and is hence underactuated. The amount of control is very simple in this design, since it relies on the springiness of the TPU, to return to an open grip state. One of the prototype from Crooks et al. can be seen in the top left corner of figure 6.

Another two-finger gripper was made by Godaba et al. Their gripper includes a variable stiffness mechanism achieved by actuation of pneumatic "stiffness pouches" resulting in actively controlled grasp modes. Though the stiffness mechanism itself is actuated by pneumatics, the finger actuation is done directly with a servo motor [6]. This gripper is seen at the bottom of figure 6.

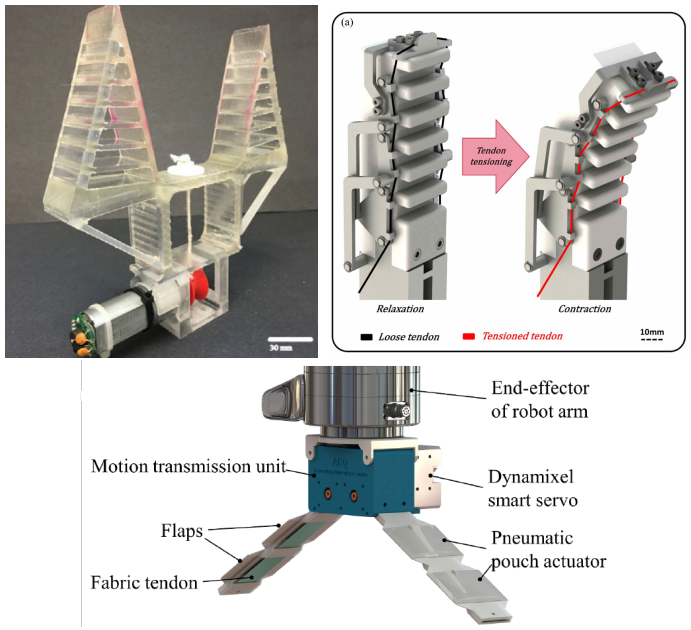


Figure 6: **Top-Left**: Screenshot of the FinRay®effect gripper by Crooks et al. [5], **Top-Right:** Antagonistically induced variable stiffness mechanism of a servo driven silicone finger from Suo et al. [7] (screenshot), **Bottom:** Two-finger servo actuated gripper with variable stiffness pneumatic pouches [6] (screenshot)

In our design task, the weight and size of both the mechanical and electrical parts play a vital role. Moving from a two-finger design to a three or even five-finger design will potentially increase the implementation complexity and form-factor. However, we will still take inspiration from these multi-finger designs, since the principles applied are still useful.

The first example of multi-finger mechanisms using underactuation comes from Ma et al. [8]. Together with the Yale OpenHand project they made an affordable 3d printable gripper, using only one motor. It is based on a simple differential pulley force distribution system. As they put it: "Underactuated hands have been shown to improve the generality of simple grippers by adaptively conforming to the surface of objects without the explicit need for sensors or complicated feedback systems." [8]. The hand has four fingers with two joints each creating eight degrees of freedom. These eight joints are all pulled at the same time and the pulley mechanism passively distributes the load among the fingers, creating a grasp that conforms to the objects being gripped. This gripper can be seen to the left in figure 7.

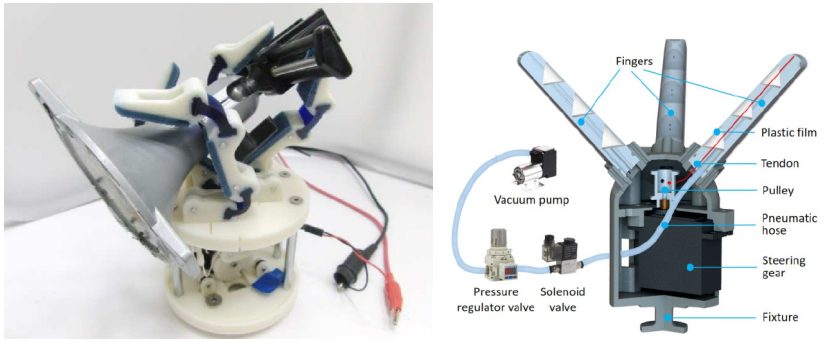


Figure 7: **Left:** Yale OpenHand Model T grasping a irregularly shaped object [8] (screenshot) - the fingers conform to object due to the under actuation pulley load distribution system, **Right:** Combined pneumatic and tendon driven three-finger gripper. Fingers are 3D printed from TPU, and tendons are driven directly by a servo motor. [9] (screenshot)

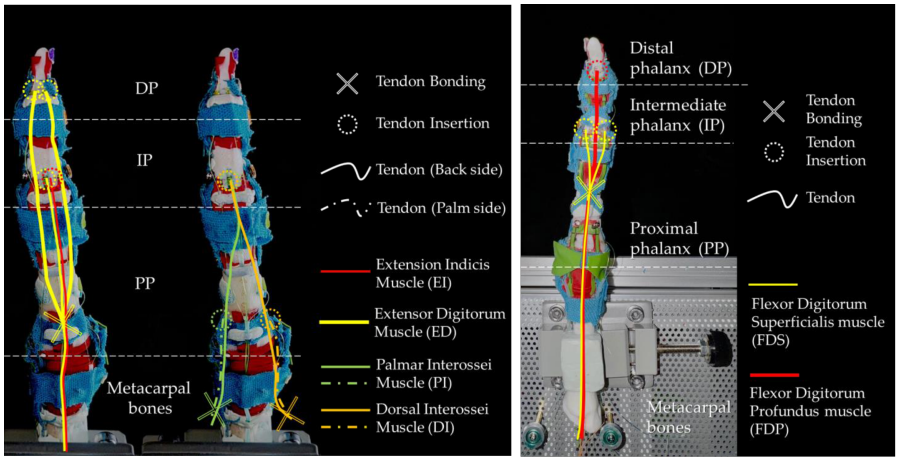
Zhu et al. [9] made a combined pneumatic and tendon driven tri-finger gripper using TPU as the structure of the finger and a servo motor as the tendon driver. This gripper illustrates the difference in grasping performance and characteristics, between pneumatic and servo actuation. Furthermore, it shows a fully 3D printed TPU finger design. See the prototype to the right in figure 7.

Sou et al. [7] made a variable stiffness (antagonistic tendons exoskeleton) tri finger underactuated gripper with silicone fingers. This design showed an improvement in finger rigidity, when using the antagonistic exoskeleton. However, the tendon drive exoskeleton increases integration complexity and

part count. The design can be seen in the top right corner of figure 6.

Moving onto grippers with more fingers, that mimic the human morphology - so called anthropomorphic grippers. We start with the most elemental form - an almost exact replica of a single human finger.

The human finger structure has been replicated by Liu et al. [10] and can be seen in fig. 8. This finger is a very direct form of bio-inspiration, to the point where it could almost be called bio-replication. The finger is actuated using a shape memory alloy wire, and ligaments are replicated using fabric.



(a) (b)

Figure 8: Liu et al. [10] (screenshot): Illustration of insertion of the muscle tendons: (a) tendon of extension muscles (view from back side); (b) tendon of flexion muscles (view from palm side). This is an example of bio replication.

Li et al. [11] have created a highly underactuated antagonistically tendon driven gripper. This gripper mimics the human hand, and includes a specific tendon guiding system. They proposed different control sequences (see figure 9) to the flexor/extensor motors, that each result in a different hand gesture. This is a way of achieving a sort of control of the actuation, while still getting the benefits of the under actuation. For guiding the tendons, they use miniature pulleys hence eliminating a lot of friction - while increasing the mechanical build complexity. This differs from the SpiRobs strategy, which instead exploits the friction for controlling the fingers/tentacles. The tendons are connected to a common axle, one for the flexor and one for the extensor. This axle is directly rotated by servo motors. Between the servo axle and the hand itself, springs are place to ensure a coordinated movement of each finger - this however increases the length of the base of the gripper. This gripper can be seen in figure 9.

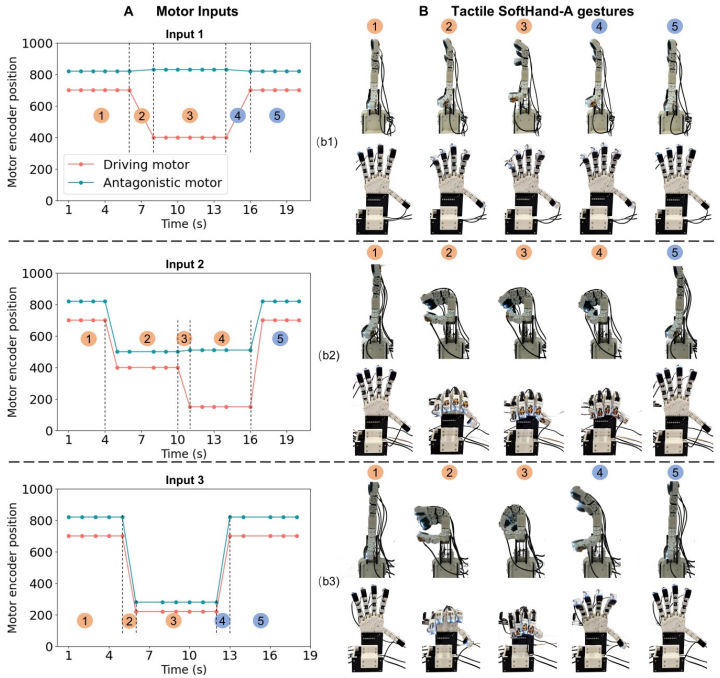


Figure 9: Motor control inputs of Tactile SoftHand-A from Li et al. [11] (screenshot). It can be seen that different actuation curves results in different grasp modes.

Bombara et al. [12] made a soft silicone gripper, also using antagonistically actuated flexor/extensor tendons. The special thing about the proposed gripper is the twisted tendon actuation. This actuation type allows for very little space to be used for DC motor, since the gearing is inherent to the twisted string, instead of being present as gears in a servo motor. However, the twisting part of the tendons usually also need a section to rotate freely in, thus using extra space. By using a single silicone mold for the entire hand, they ensure a soft and high friction contact area no matter the grasp modes. The question of structural integrity (stiffness) is solved by using PTFE tubes (for tendon guidance) and as the "bones" of the hand - resulting in a constant stiffness.

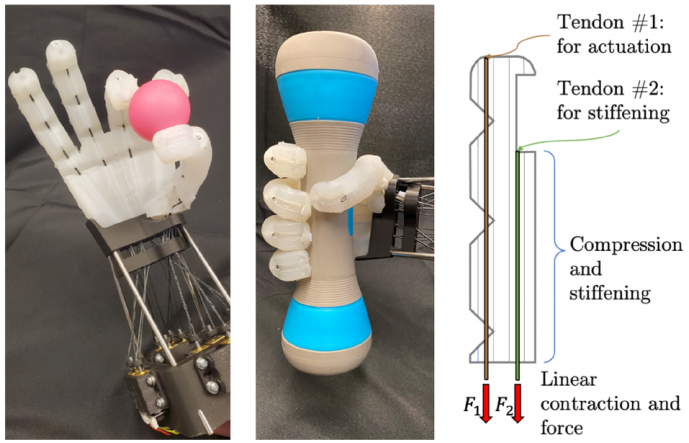


Figure 10: Tendon setup for the soft robot hand made of silicone proposed by Bombara et al. [12] (screenshot)

Common for all of these grippers is the internal placement of the actuators. Meaning that the motors are placed inside the gripper itself. This increases the weight of the gripper substantially, and should ideally be tackled differently in this project, since the weight is important.

Additionally, most of the mentioned grippers contain a tendon distribution system, that takes up quite a lot of space (and thus also weight). By tendon distribution system, we refer to the space between the actuators and the fingers, that often constitutes itself as a "bulky" box at the base of the grippers. Again, due to weight constraints, this also seems like a challenge that we have to deal with differently in this project.

The reviewed grippers demonstrate varying degrees of bio mimicry, ranging from direct morphological imitation to the transfer of abstract biological concepts. For the reviewed grippers, the latter approach results in reduced mechanical complexity, which is a critical factor for the weight and size constraints of this project.

Figure 11: Diagram showing the balance and tradeoff between rigid/soft robots.

Another challenge identified is the balance between a soft and a rigid structure. This is the tradeoff that is inherent to the original motivation for using soft robots - the ability to conform to objects automatically. This ability directly results in a decrease of structural strength and performance. As we have seen from the previous grippers, one way of solving this challenge is by the use of variable stiffness, like pneumatic pouches or, more relevant to this project; antagonistic tendon movement.

With the principles and concepts of these previously developed grippers, we will now begin the development of our gripper. We start at the conceptual design phase in the following section.

4 Conceptual Design

In this chapter, we translate the problem definition and requirements established in section 2 into a concrete robotic system, while drawing inspiration from designs seen in the previous section on other grippers. Having selected the tendon-driven strategy over the pneumatic approach in section 2.3 , the goal is now to generate specific mechanical and electrical solutions that uphold the requirements listed in table 2.

4.1 System architecture

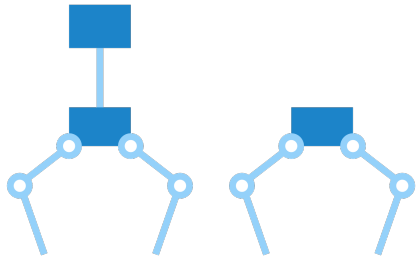


Figure 12: **Left:** An architecture where the actuator is placed externally. **Right:** An architecture where the actuator is placed internally to the end-effector

Designing a robotic gripper requires a holistic view of both the electronics and mechanics and how these interact. The system is a chain of subsystems going from control, activating drive electronics, actuating a motor that converts electrical energy to mechanical force, which is then guided down to a physical interface with the environment.

We have two main architectures we can choose from:

1. **Displaced Motor:** Fig 12 (Right side). In this configuration, the "heavy" actuation components are placed at the base of the robot or on a static platform. The force is transmitted to the end-effector via a transmission system (e.g., Bowden cables). This system saves weight on the gripper but has additional resistance from the transmission system. In a pneumatic or hydraulic system this resistance is almost negligible but in a tendon-sheath mechanism, the resistance is considerable as it is mainly governed by the Capstan equation ($F_{\text{output}} = F_{\text{input}} \cdot e^{\mu\theta}$) [13] which is an exponential relationship. The resistance can also introduce hysteresis to the system, due to the effects of static and kinetic friction.
2. **Motor on End-Effector:** Fig 12 (Left side). In this configuration, the motors, and potentially the drive electronics, are mounted directly on the wrist of the robot arm. This eliminates the resistance and hysteresis, but adds additional weight of the motor + electronics to the gripper. However, if the end-effector was to be used for other robot arms than the NRT-Lab arm, this would highly favor this architecture.

The benefits of having a motor on the end-effector made us look into lightweight motors. Picking a motor is a balance between weight, cost, power and performance. The ultra lightweight and compact motors, usually contain a limited amount of internal control and down-gearing, while also being quite costly.

An example could be the "Series 0816 ... SR" DC motor from Faulhaber [14], which are extremely lightweight (4.5 g), but with a low stall torque of 0.11 Ncm and a very high price point it is beyond the budget of this project.

We therefore went with the displaced motor architecture, since then we could use cheap servos with more power, integrated down-gearing, encoders, and internal control that could speed the development up.

4.2 Subsystems

As described in the previous section a tendon driven gripper generally consists of control $\rightarrow$ drive electronics $\rightarrow$ electromechanical actuation $\rightarrow$ mechanical transmission $\rightarrow$ and the physical interface with the environment (the fingers).

These parts of the systems can be put into separate subsystems, allowing us to make different conceptual designs of the subsystems, and put them back later in a modular way.

Therefore we made five subsystems:

1. **Fingers:** This is the part that interacts with the gripped object. As specified in our prior work, this design concept leans up against the human hand, using tendons to actuate fingers.
2. **Finger mount:** The finger mount connects the end of the robot arm, the transmission line and the fingers
3. **Transmission line:** This is the transmission line that connects the actuator and finger mount.
4. **Actuator:** This subsystem involves the motors and the appropriate mechanisms for linking the motor rotation to tendon pull.
5. **Control:** The control subsystem involves choosing a control strategy - eg. how are the tendons moved relative to each other, and what variables control the movement (PWM, Current, position etc.).

We are now ready to make conceptual designs of the subsystems that uphold the requirements listed in the previous chapter. We start with the fingers.

4.3 Fingers and Bio-inspiration

To make the fingers, we use the bio-inspired design method. In our prior work we searched for grasping in nature, and found hands and tentacles and elephant trunks as our primary inspiration. We will now understand and transfer this into a design, that upholds the requirements from table 2.

Our design process initially focused on the human finger as the primary inspiration for the gripper's structure and function. We chose this starting point because of its mechanical simplicity. While nature offers other flexible examples, like elephant trunks or octopus tentacles, we initially avoided these directions, fearing that mimicking their complex, continuous movement would result in a gripper that was too heavy and difficult to control. However, during the prototyping phase, we discovered a hybrid solution: we could utilize the simple, discrete joints of a finger-like structure to emulate the wrapping shape and behavior of an octopus tentacle. The following section details the iterative process that led to this insight.

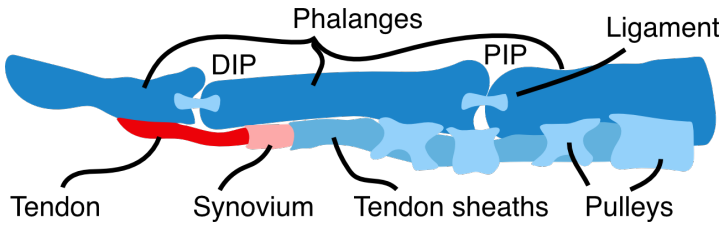


Figure 13: A simplified schematic of the human finger. DIP is short for distal interphalangeal joint, and PIP is short for proximal interphalangeal joint.

Human finger-joint actuation

Figure 13 provides a simplified schematic of the anatomical components relevant to our design. The human finger consists of phalanges (bones), tendons that actuate the joints, synovium which functions as a lubricant, tendon sheaths that encase the tendons, and pulleys that anchor the sheaths to the bone. When tension is applied to the tendons, the force is transmitted to the phalanx. The pulleys constrain the tendon close to the bone, converting this linear tension into a rotational bending moment at the joint [10].

Human joints

Human finger joints are synovial joints, relying on fluid for lubrication and ligaments for bone-to-bone stability. While Liu et al. [10] successfully replicated this structure, the resulting design (Fig. 8) demonstrates the downsides of a strict 1:1 bio-replication strategy. The intricate assembly of individual components, mimicking every ligament and pulley, drastically increases both the fabrication complexity and the overall weight of the gripper, making this approach unsuitable for our goal of a lightweight gripper.

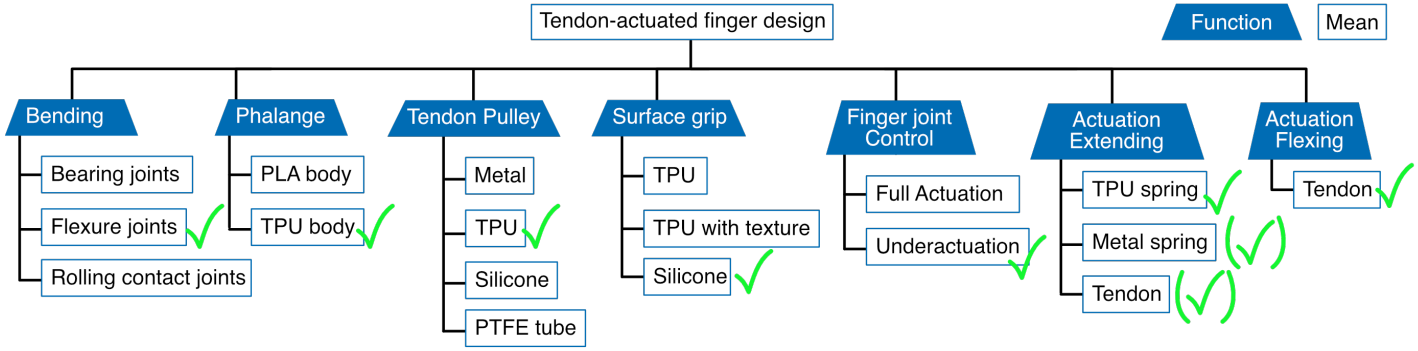


Figure 14: Function means analysis for the fingers. For clarity, the chosen means are already marked.

Functions

In contrast to Lui et al. [10], we abstract the finger functions to achieve a lightweight design. By mapping the biological functions into the function-means tree in Fig. 14, we identify the optimal means to mimic the finger's functionality. For clarity, the chosen means are already marked. The rationale behind these design choices is detailed below.

Bending

The fundamental function of the finger joint is articulation. To achieve this, three primary means were considered. First, an anatomically accurate approach like Liu et al. [10], utilizing rolling contact joints with fabric ligaments. While bio-mimetic, the complexity of this assembly contradicts the goal of a lightweight design. Second, bearing joints, which are standard in robotics due to their low friction and large range of motion. However, the metallic components impose a significant weight penalty. Finally, flexure joints. Although these introduce elastic resistance and have a finite range of motion, they are superior for lightweight applications. This is because they rely on material compliance (elastic deformation) rather than complex mechanical assemblies.

Phalange

While the joints provide the necessary compliance to conform to objects (as discussed in Bending), the phalanges must address the opposing requirement: rigidity. As highlighted in the review of previous grippers, a purely soft structure risks lacking the strength to transmit force effectively or support internal components like pulleys.

To minimize weight and assembly complexity, we aim for a monolithic structure. Furthermore, to maintain the intrinsic benefits of a soft gripper, such as safety and adaptability, we avoid integrating completely rigid materials like metal or hard plastics. Consequently, we rely on a single soft material for the entire structure, accepting that this design choice involves a trade-off regarding maximum structural strength compared to hybrid designs.

Tendon Pulley

The tendon pulleys keep the tendon close to the bone so it doesn't pull away when the finger bends. This ensures the pulling force is used effectively to curl the joint. To keep the design simple to make, we are using the same material for the pulleys as the finger bones, printing them in a single piece. While some prior works, such as Li et al. [11], employ PTFE tubes to minimize friction, we prioritize the mechanical simplicity of a monolithic design over the low-friction benefits of PTFE inserts.

Surface Grip

Human skin balances durability with the softness to deform under load. Cutkosky highlights that this surface deformation is essential for defining the contact conditions in robotic grasping [15]. To emulate this, we favor silicone over TPU for its superior softness and resulting grip. However, this choice introduces a durability trade-off: molding silicone onto the underlying TPU structure creates a multi-material interface that may be weaker and more prone to delamination or wear than a single-material TPU part.

Finger joint control

We can either fully actuate and control every joint, or choose under actuation for the fingers. Under actuation is the concept of having less degrees of actuation, than degrees of freedom. If we had full control of the fingers, each joint would require a tendon back to a separate motor. Two fingers with two degrees of freedom quickly becomes four motors, which would not comply with our requirements of simple control, and it would be harder to fit the transmission inside the soft arm.

Flexing and Extending

Human fingers employ antagonistic tendon pairs to actively control both flexion and extension. To minimize weight and control complexity, our first design iteration eliminates the extensor tendon, relying instead on passive extension. This strategy utilizes the elastic potential energy stored in the flexure joints during flexion to return the finger to its neutral position.

It is important to note that substituting active extension with passive material elasticity involves a functional trade-off. In an octopus-inspired morphology, relying solely on material stiffness can result in unwanted movement patterns or hysteresis. Furthermore, it does not resolve the conflict between compliance and strength (Fig. 11). Due to these limitations, we eventually developed a second iteration of the design that reintroduced an active extension tendon, a modification that will be discussed further in the embodied design chapter.

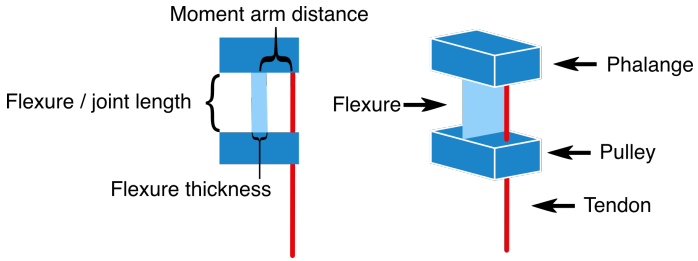


Figure 15: Principle structure of the finger joints

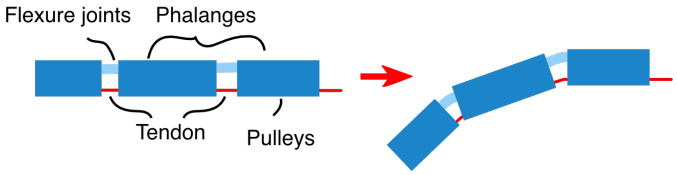


Figure 16: Principle function of finger joints

Other flexures and FACT (Freedom and Constraint Topologies)

Based on the functional requirements, we developed a principal structure consisting of rigid segments ("phalanges") connected by compliant flexure joints. As illustrated in Fig. 15 and Fig. 16, the tendon runs through the rigid phalange at a distance d from the neutral axis. When tension is applied, it generates a moment that bends the flexure.

To predict the behavior of these joints, we model the flexure as a cantilever beam (blade flexure) subjected to an end moment. The fundamental relationship for beam deflection is $\theta = -\frac{M \cdot L}{E \cdot I}$ [13]. By substituting the moment of inertia for a rectangular cross-section ($I = \frac{w \cdot t^3}{12}$) and the moment generated by the tendon ($M = F \cdot d$), we derive the governing equation for the joint angle θ:

$$\theta = \frac{d \cdot F \cdot L \cdot 12}{E \cdot w \cdot t^3}$$

Where F is the tendon pulling force, d is the moment arm (distance from flexure center to tendon), L, w, t are the length, width, and thickness of the flexure, E is the material's Modulus of Elasticity.

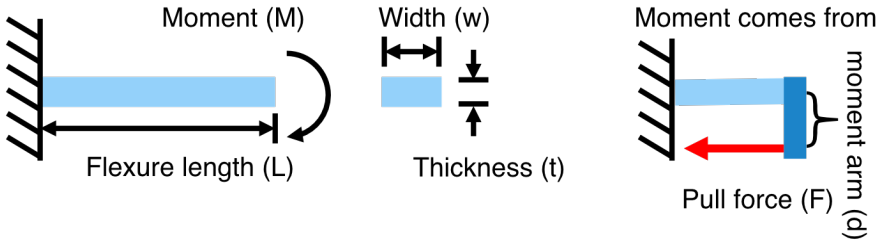


Figure 17: Beam bending analysis of a blade flexure

This relationship highlights that the joint's stiffness is cubic with respect to thickness (t). Therefore, tuning the wall thickness is the most sensitive parameter for adjusting the finger's bending profile.

Flexure Topology and FACT Theory

The beam analysis above approximates bending stiffness but does not account for off-axis movements, such as twisting or side-to-side wobbling. To ensure the finger bends only in the intended plane, we utilized the Freedom and Constraint Topologies (FACT) method developed by Hopkins [16].

FACT analyzes how specific geometries constrain degrees of freedom. A simple blade flexure (Fig. 15) creates constraint lines that resists compression but offers little resistance to torsion along its vertical axis. To improve stability, we considered alternative topologies shown in Fig. 18:

- **Cross Hinge:** intersecting blades that eliminate the parasitic center shift of the rotation axis.
- **Living Hinge:** A thin, localized reduction in material.

While FACT assumes infinitely stiff and thin members—an idealization not fully achievable with soft materials, it provides the theoretical basis for selecting a joint geometry that maximizes bending compliance while resisting twisting moments during a grasp.

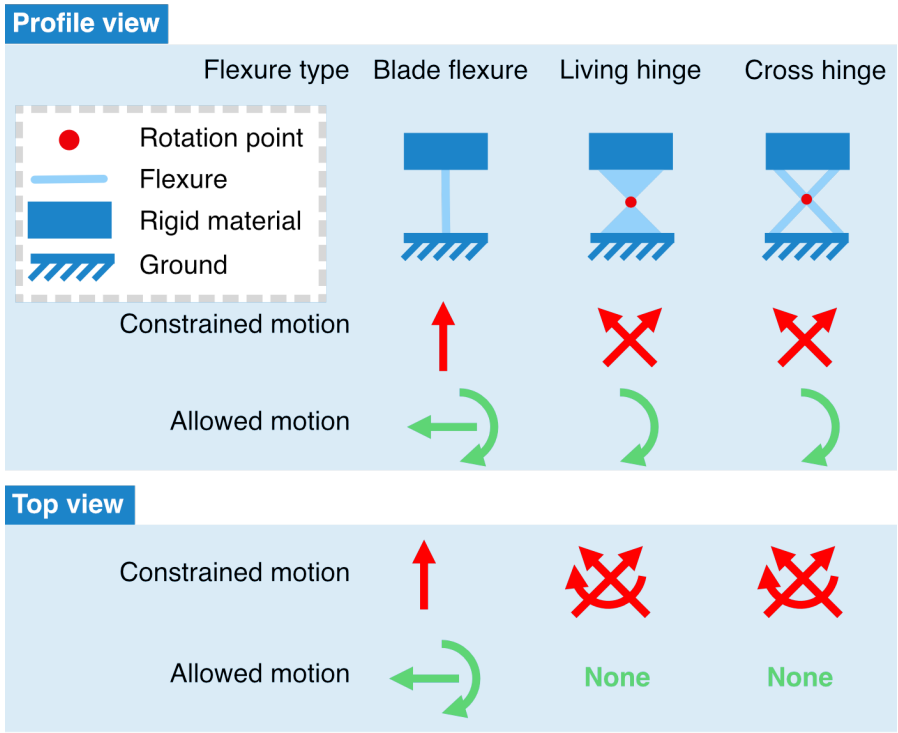


Figure 18: Three different flexures allowing rotational motion. Assumptions: Infinitely stiff and thin flexures.

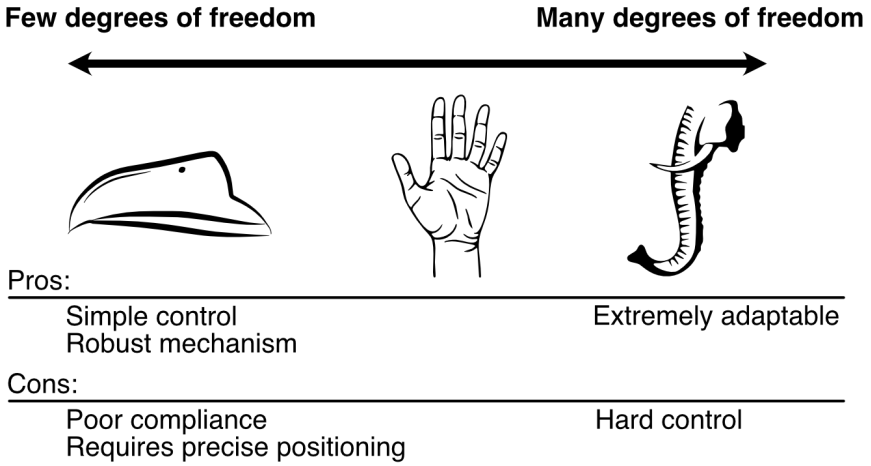


Figure 19: **The Degrees of Freedom (DoF) Spectrum in Gripper Design Left:** (Bird's beak) 1-DoF systems prioritize mechanical robustness and simple control but offer limited compliance. **Right:** (Octopus) Many degrees of freedom provide high adaptability but increase control difficulty. **Middle:** The human hand represents a versatile intermediate, capable of multiple distinct grasp modes, due to the multiple fingers.

Bio-inspired morphology

As summarized in Fig. 19, the number of degrees of freedom (DoF) dictates a gripper's functional trade-offs. Low-DoF biological mechanisms, such as a bird's beak, provide rigid, precise pinching. Conversely, high-DoF structures, such as tentacles, excel at conforming to and wrapping around objects. The human hand occupies a versatile middle ground.

To achieve our goal of a compliant gripper, we prioritize the wrapping capability found in nature's softest manipulators. Therefore, we move away from the rigid kinematics of the human hand and mimic the morphology of an octopus. Consequently, our conceptual design adopts the flexible, multi-jointed structure shown in fig. 20.

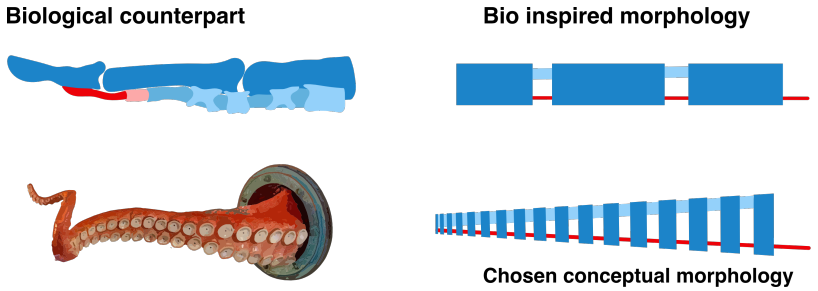


Figure 20: Bio inspired morphology

With the morphology and actuation strategy defined, the remaining challenge is anchoring these soft components to the rigid robotic system. The following section details the design of the Finger Mount, which serves as the interface between the soft gripper and the rigid robot arm.

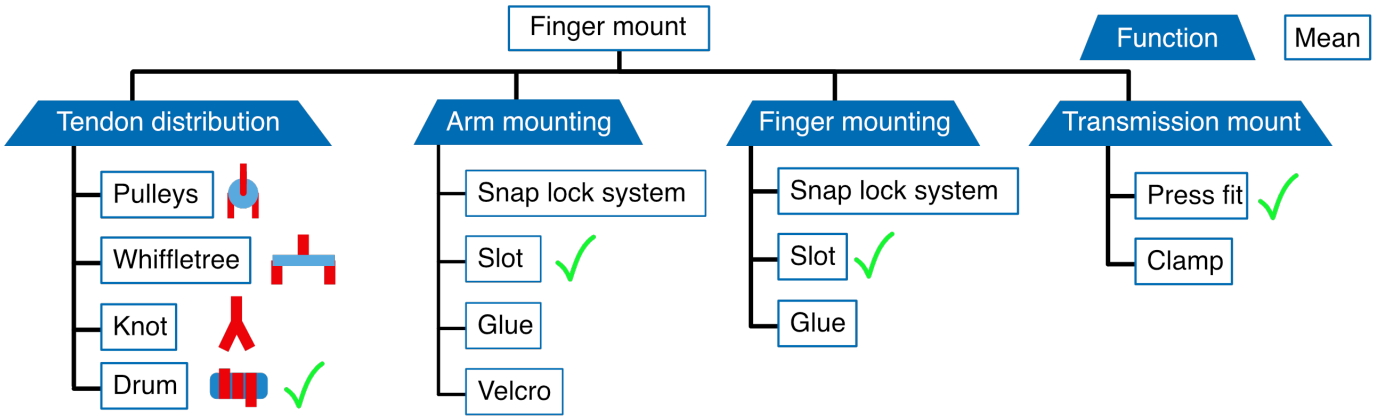


Figure 21: Function means tree for the finger mount

4.4 Finger mount

Now that we have made conceptual designs for the fingers, we now have to mount them, and connect them to the rest of the system. The finger mount has some functions it needs to perform. It has to be mounted on the end of the soft arm, it has to take the input of the transmission and distribute it to the fingers of the gripper. These functions are made into a function means tree in fig. 21.

Necessity of the finger mount

We will now argue on the necessity of the finger mount, as opposed to having fingers directly on the arm. If we had one transmission per finger tendon (and therefore did not need tendon distribution), we would have a bulky transmission that might not uphold the requirement from table 2, about fitting inside the NRT Lab soft arm.

The tendon distribution mechanism could be a simple knot as fig. 22 shows. This knot, however, needs some travel distance, from the fingers to the transmission, as it cannot go into the fingers. A quick estimation from one of our later prototypes was that 60 mm of travel distance was required. It is therefore necessary with this a separate finger mount subsystem, as we cannot mount the transmission directly to the fingers.

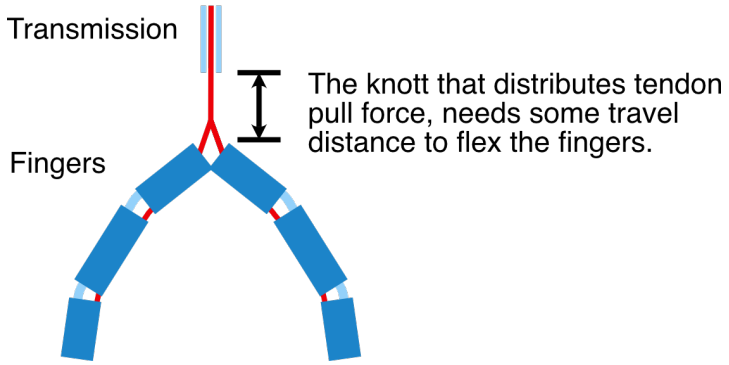


Figure 22: Tendon distribution travel problem

Tendon path considerations

The tendon path is the route the tendon follows from its origin at the motor spool to its termination point at the fingertip. Along this path, the ten-

don passes through several regions of the system: first the actuator section, where tension is applied, then the soft robotic arm, where the tendon is guided internally; and finally the finger mount and the fingers themselves, where the applied tension is converted into bending motion.

For accurate and predictable actuation, the tendon path must be rigid from the motor spool up to the finger root, ensuring that any tension produced by the motor results solely in finger motion. If this section flexes or deforms, part of the tension is absorbed before reaching the fingers, leading to lost force and hardening the control task. Therefore, to prevent energy loss, the finger mount is designed as a rigid component.

Tendon distribution

The most critical function of the finger mount is tendon distribution, splitting the single input force from the actuator to multiple fingers while minimizing friction and weight.

We initially evaluated passive distribution mechanisms from the Yale OpenHand project, specifically the floating pulley and whiffletree (visualized in the top row of Fig. 24).

While Ma et al. [8] suggest these mechanisms improve grasp adaptability by allowing fingers to close independently, our analysis indicated that they do not address the fundamental "travel distance" problem described in the previous section. These linear mechanisms require the same extensive housing length to accommodate the tendon stroke, failing to solve the packaging constraints of the soft arm.

To resolve this, we selected the rotational drum mechanism (Fig. 24, bottom). By wrapping the tendon around a spool, this design converts the necessary linear travel (60 mm) into rotational motion, drastically reducing the component's footprint. Physical prototyping confirmed the superiority of this approach: a linear housing designed to accommodate the "knot" method weighed approximately 25 grams and was significantly bulkier than the compact drum prototypes. Thus, the drum offers the optimal balance of compactness and weight reduction.

Drum orientation

The orientation of the distribution drum (horizontal vs. vertical) dictates the gripper's overall form factor and efficiency. This choice impacts three critical areas:

- **Finger angle:** The tendon should exit the drum tangentially and align directly with the finger. Any deviation creates friction points governed by the Capstan equation ($F_{out} = F_{in}e$). While adding extra pulleys could correct this alignment, they would add unwanted weight.
- **Finger distance:** A vertical drum requires more space for mounting brackets and tendon routing. This forces a wider distance between the fingers compared to a horizontal setup.
- **Finger quantity:** As the tendon for the finger, should align tangentially with the drum, the vertical drum option can have many fingers around it without sacrificing friction, whereas having more multiple fingers on the horizontal drum, would sacrifice friction, if the fingers were to be in in the same plane of orientation.

Accommodating Two Tendons for the Second Iteration Fingers

To support the active extension introduced in our second design iteration, the tendon distribution mechanism was adapted to house two independent tendon lines. While this could be achieved using strictly horizontal or vertical drums, we utilized a hybrid combination of one vertical and one horizontal drum (Fig. 23). This configuration successfully accommodated multiple fingers within the compact housing, albeit at the cost of slightly increased friction due to some additional tendon routing.

Quantity of Fingers

The design of the end-effector requires a balance between finger count and weight. Increasing the number of fingers expands the range of available grasp modes, and makes a more secure grasp, it also adds weight. In our prior work, we found that three fingers are typically required to fully constrain an object, however, this assumes the fingers only have point contacts. A stable grasp can be achieved with only two fingers if the contact surface is sufficiently large. These wider contact areas allow the fingers to resist mo-

ments, thereby providing the necessary stability while reducing the reduced finger count.

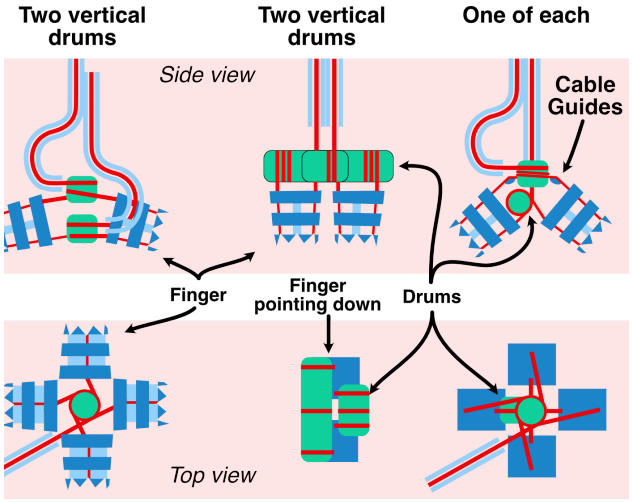


Figure 23: Drum orientation for two tendons.

Mountings

The finger mount serves as a central hub connecting three distinct subsystems. We evaluated the fastening methods for each interface based on weight, structural integrity, and ease of assembly/disassembly (serviceability).

- **Finger Interface:** As detailed in Appendix I, we explored various methods to secure the fingers to the mount. Based on our evaluation, the slot mechanism was selected. This design provides a lightweight connection allowing for rapid prototyping of fingers, and in the final prototype, a more modular solution.
- **Transmission Interface:** The connection to the transmission line was selected using a decision matrix (Appendix J), again prioritizing the potential for modularity by using a slot.
- **Arm Interface:** For the connection to the arm's rigid end-plate, we opted for mechanical fasteners (screws). Due to the circular cross-section, implementing a slot mechanism would be difficult to align. Additionally, screws provide a rigid fastening method, ensuring better stability under load.

Functional surfaces

Functional surfaces are the specific geometries required to interact with other components. For the finger mount, these essential interfaces are:

- The drum interface
- The finger slots
- The transmission interface
- The arm interface

Any material outside these regions that does not contribute to structural integrity is considered excess mass. To systematically remove this dead weight while maintaining rigidity, we utilized Generative Design (Autodesk Fusion 360).

In this workflow, the functional surfaces were designated as "Preserve Geometry," while the tendon paths and rotating drums were defined as "Obstacle Geometry" to prevent material generation in these zones. The software then synthesized the optimal structure to connect these surfaces using iterative Linear-Static FEA [17]. A key advantage of this approach over traditional topology optimization is the integration of manufacturing constraints; specifically, we enforced a minimum strut thickness of 3 mm to ensure the resulting geometry remained stable during and after FDM printing.

We have now discussed the design concepts of the finger mount. In the next section we will move on the the connecting subsystem between finger mount and actuator: transmission.

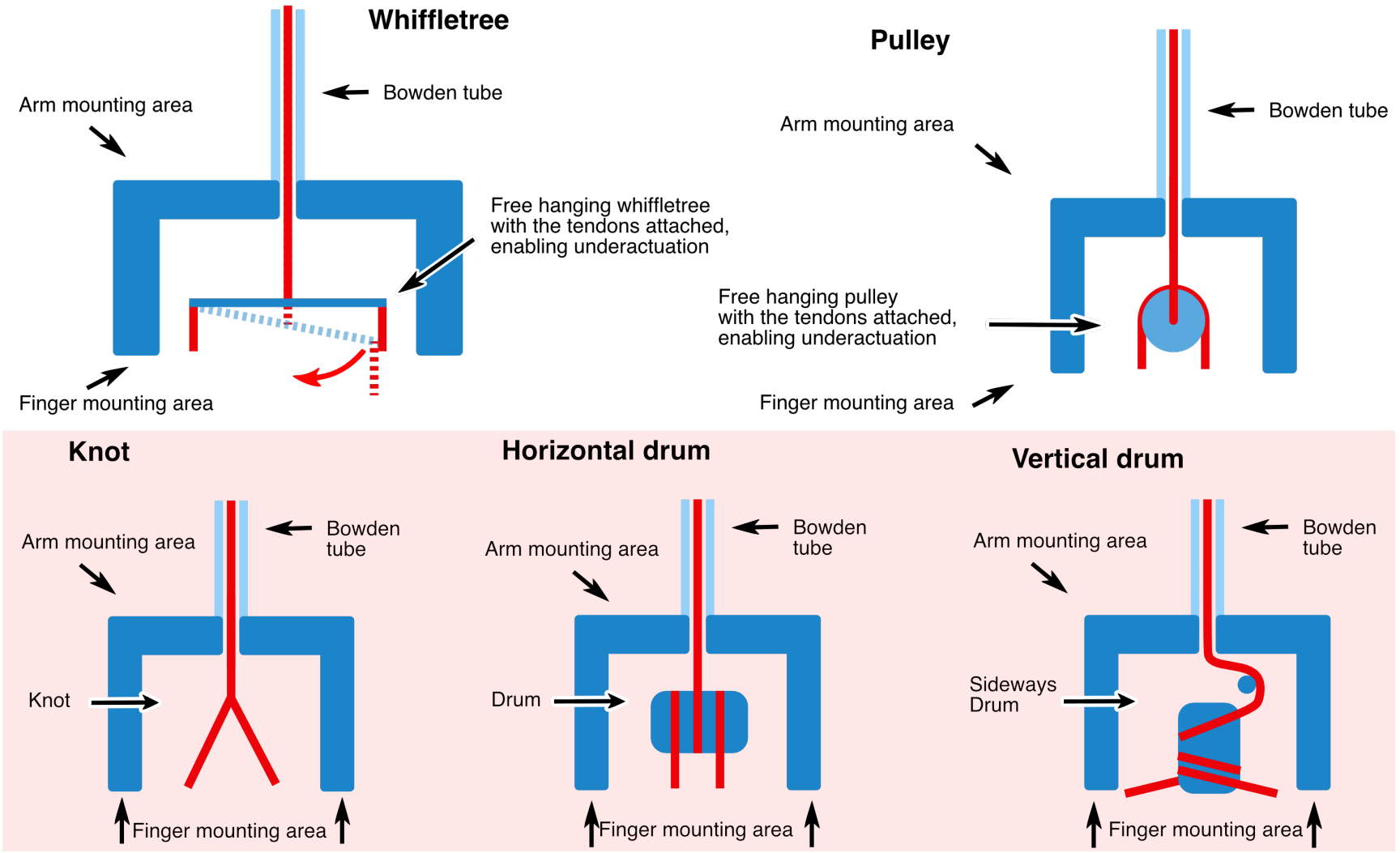


Figure 24: Different means for tendon distribution

4.5 Transmission

The purpose of the transmission is to transmit tendon pull from the actuator to the finger mount, where it will be distributed to the fingers. The primary requirement is to deliver this force with minimal energy loss to ensure the smooth operation of the gripper (as described in the previous "tendon considerations" section). It is therefore necessary to analyze the forces governing the transmission to minimize resistance. This mechanism is commonly referred to as a tendon-sheath mechanism.

The Capstan Equation

The governing principle of the transmission is the Capstan equation [13], which models the tension loss due to friction along a curved path:

$$\frac{F_{in}}{F_{out}} = e^{\mu\theta}$$

Where F_{in} is the actuator force, F_{out} is the resulting force at the finger, μ is the coefficient of friction, and θ is the total cumulative curvature (in radians). To maximize efficiency ($F_{out} \approx F_{in}$), we must minimize both μ and θ. However, as the curvature θ is largely dictated by the pose of the soft arm, our design focus must remain on minimizing the friction coefficient μ.

Static friction inside the arm also introduces hysteresis; the static friction threshold must be overcome before the tendon inside the transmission can move relative to the sheath. This results in a "dead-band" behavior that is detrimental to precise control, further emphasizing the need for low-friction materials.

Axial Stiffness

In addition to friction, tendon elongation causes control issues. If the tendon has low axial stiffness (high elasticity), it will stretch before transmitting motion to the finger. This creates a spring-like hysteresis where the actuation of the finger is dependent on the loading history of the wire. Therefore, the ideal tendon must possess high axial stiffness (to prevent stretch) while maintaining high bending flexibility (to move smoothly through the sheath and the drums).

Component Selection

For the sheath, a PTFE (Polytetrafluoroethylene) tube was selected due to its flexibility and extremely low friction. PTFE is widely known as a "self-lubricating" solid, boasting a friction coefficient of approximately $\mu \approx 0.02 - 0.2$ against polished steel [18]. We utilized standard tubing common in 3D printing (2 mm ID / 4 mm OD) ensuring both low cost and high availability.

For the tendon, we selected an Ultra High Molecular Weight Polyethylene (UHMWPE) wire. This material exhibits high axial stiffness (negligible length change under load) while remaining flexible in bending. We selected a 1.5 mm diameter UHMWPE wire; this diameter fits with sufficient clearance inside the 2 mm PTFE tube, minimizing the risk of binding. Additionally, the thicker profile (compared to standard fishing line) provides a larger surface area, reducing the likelihood of the tendon cutting into the soft finger structure under load.

We have now defined the conceptual design of the transmission and will move on to the next subsystem, which revolves around creating the tension of the tendons.

4.6 Actuator

The purpose of the actuator module, is to provide motion and force, which is transmitted to the fingers. Since we are making the actuator module external to the gripper, we don't really have any weight or size constraints which results in quite a lot of freedom when designing it. We will now go through some of the considerations we had concerning the actuator module.

In figure 25 we see an abstraction that helps us to understand the requirements of the actuator module. Everything located inside the red box, is defined as the actuator module. The control and feedback arrows will not be explained here, but in the next section (control section 4.7).

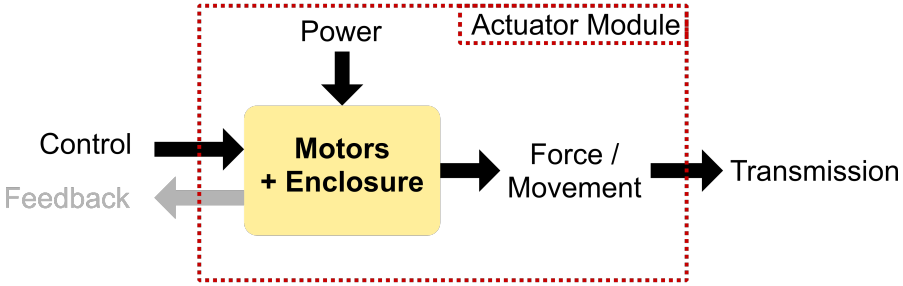


Figure 25: Abstraction of the actuator module functions. Control and feedback arrows are related to the control subsystem. The force is related to the transmission.

This leaves us with the motors, force interface to the transmission (tendons) and the power supply. In figure 26 we see a function-means tree with these functions. We will now explain these chosen concept choices.

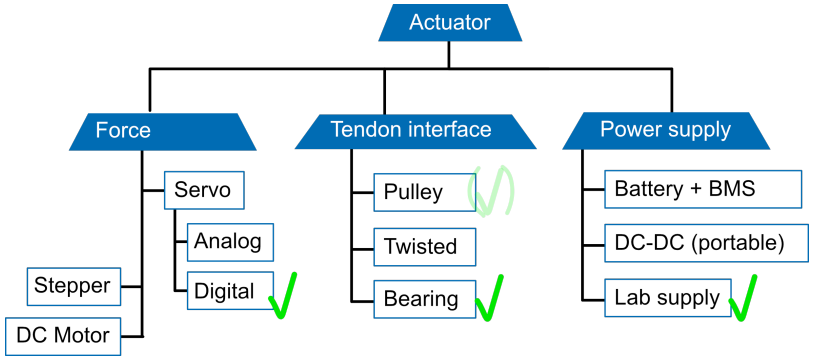


Figure 26: Function means tree for the actuator subsystem

Force and movement

As mentioned in section 2.3, we have chosen to use DC motors as the actuating basis of the gripper. As can be seen in the function means tree (figure 26), we have the choice between using "pure" DC motors, servo motors or stepper motors. Here "pure" DC motors are defined as a DC motor without any internal control, gearing and such.

The advantages of using "pure" DC motors are the low weight, size and high customizability. On the flip side they often don't come with any encoders or internal control logic, which would require us to spend resources on the design of these parts ourselves. This is outside the scope of the project, and we will therefore not be using "pure" DC motors.

Both stepper motors and servo motors rely on some type of control electronics. Stepper motors use open-loop control and excel at very predictable and repeatable motions, by rotating in discrete steps. Servos use closed-loop control and gearing and are therefore usually better suited for achieving a compliant feel. [19].

These thoughts lead us to the choice of using servo motors. This choice resonates well with the chosen system architecture (external actuator subsystem), since weight and size is then less important. A future improvement could be the investigation of the use of "pure" DC motors, to enable incorporation of internal actuation in the gripper.

When choosing a servo motor for an external setup we need to balance performance and cost. An overview of a few different types of readily available servo motors in our price range is presented in table 3.

The candidate motors are sorted by increasing stall torque. The stall torque parameter is important to look at, since it is often listed in datasheets and it gives us an indication of the amount of force we can expect the motor to output. Stall torque is not a measure of the amount of torque to be expected under continuous operation, but a measure of the torque output at stalling conditions [20]. This means that we need to consider how long the forces are applied and needed. Eg. if we merely need to overcome some static friction for a burst of movement, it is okay to require torques close to the stall torque. However, if we need a constant high torque for eg. holding the grip, we need to stay below the stall torque, to ensure the continuous operation of the motors.

Model	Price [€]	Weight [g]	Stall Torque [Ncm]	Supply [V]	Control level
TowerPro SG90	5	9	11-15	4.0-6.0	PWM signal (angle), internal analog control
XL330-M288-T	40	17	42-60	3.7-6.0	Advanced internal MCU control, (current)
Towerpro SG-5010	10	47	54-64	4.8-6.0	PWM signal (angle), internal analog control
XC330-M288-T	131	23	69-110	3.7-6.0	Advanced internal MCU control (current)
ST3020	24	99	145	6.0-14.0	Internal MCU control (current)
XL430-W250-T	43	57	100-150	6.5-12.0	Advanced internal MCU control

Table 3: Comparison of different servo motors that we considered during servo motor research.

The other performance factor that must be considered, is the "control level" - meaning the amount of internal control already implemented in the servo. In general we can split the servos into two categories: simple analog control and more advanced digital control. The first category works by using an analog circuitry to map the duty cycle of a provided PWM signal to an angle of the shaft encoder - simple and cheap [21]. However, as mentioned before, it is desirable for us to use a motor with as much internal control already implemented - both for the rapid prototyping and for the testing of control concepts. We will therefore turn our attention to the motors marked with "internal MCU control" in table 3.

We now have a choice between the brand Dynamixel (XL330, XC330 and XL430) and Waveshare (ST3020).

In our research, we found that the Dynamixel motors have a solid ecosystem of hardware, firmware libraries, testing software and documentation. Eg. the combination of the testing software "Dynamixel Wizard 2" [22] and the "Dynamixel U2D2" computer adapter [23] enables a quick working testing setup where the motors can be manually controlled and monitored. Furthermore, the Dynamixel motors support a wider range of different operation modes than the Waveshare, such as; Position mode, Extended Position mode, Current mode, Current based Position mode, Velocity mode and direct PWM (voltage, not angle) mode.

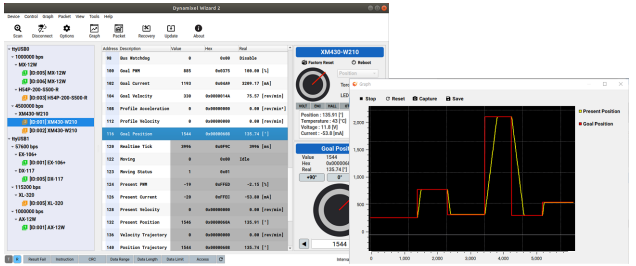


Figure 27: Screenshot of the Dynamixel Wizard 2.0 [22]

The above considerations lead us to the choice of buying a set of both the Dynamixel XL330-M288-T and the Dynamixel XL430-W250-T for testing and developing. The difference between these motors is mainly the size and hence also the stall torque rating. The XL430 servo is the bigger and stronger, and the XL330 is the smaller - we chose these two motors with different torque ratings, because in the beginning we did not know the exact amount of actuation force needed. The reason for not just picking the big (XL430) high torque servo right away, is because the small (XL330) servo is able to use current as a control parameter - which is very well suited for achieving a compliant feel of the gripper. Physically, both Dynamixel servos communicate over a TTL serial half-duplex connection.

We now have a good starting point for developing the actuator module; namely two motor options to experiment with. We will now continue to look at the interface between these motors and the tendons that actually actuate the fingers.

Tendon interface

In the last paragraph, we analyzed the choice of servo motors. Now we will move on to the interface between these motors and the transmission - specifically the interface between the rotational movement of the motor and the linear motion of the tendon inside the transmission. We will now briefly go through a couple of different concepts (seen in figure 28) capable of connecting these two motions.

The first method that we would like to highlight, is using twisting tendons. This alternative design involves attaching two parallel tendons (orthogonal) to the horn of the servo. When the servo rotates, these will twist around each other, and result in a linear displacement of the end points. This design can e.g. be seen in figure 10 (Bombara et al. [12]), where a soft robot hand is actuated using "pure" DC motors and twisted tendons. The downside to this, is the fact that you have to fit two tendons inside the transmission, and that you usually need a bit of space for the tendons to twist/untwist.

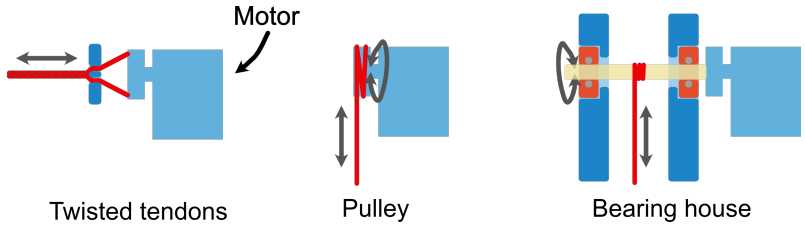


Figure 28: Principal structures of the interface between servo motor rotational movement and actuating linear movement

The second and maybe most known concept is a pulley directly mounted to the horn of the servo motor. This concept is simple and with only a single part enables the required translation of movement. The downside of this setup is the structural radial load on the axle, potentially stressing the inner structure of the motor.

To overcome the issue of stressing the servo motor with a radial load, one could further develop the pulley interface by using an external axle resting in a grounded bearing house. This ensures that the servo motor only ever sees tangential loads, and that the radial loads from the tendons get transferred to the strong(er) bearings. This second concept does introduce more parts and therefore increases implementation complexity - but because of the external actuator placement, this is acceptable.

Initially we picked the simple pulley solution, but as we will see in the embodied design, we pivoted away from the simple pulley and towards the bearings, to spare the axle of the motor from the radial stress.

Power

The last function in the actuator module is the the power supply to the motors. Since we have chosen the Dynamixel motors XL330 and XL430, the voltage of the power supply should be either 5 or 11V (see table 3). We could use a small DC-DC converter board to power the electronics, or even use a battery to make it portable. However, the main focus of this project is not a finalized product, but a first prototype, meaning that the power source will most likely be the least prioritized optimization task. This means that we will stick to using a variable lab power supply for most of the developing and testing of the gripper.

This brings the section of conceptualizing the actuator subsystem to an end. In the next section we will discuss the input and output of the actuator: control and feedback.

4.7 Control

The purpose of the control subsystem is to have a way of communicating with the actuator module. This communication enables us to develop and test the gripper. It is not the goal to have a final (physically) miniaturized control system - however this would be the obvious next step after achieving a rough first prototype. For now we will go through the different concepts of the control, which we have worked with in the project.

Hardware

The first thing to consider when designing the control system is the physical framework (hardware) that hosts the firmware that controls the gripper. This framework needs to be fast to prototype on, and ideally include proper documentation and a community-driven support system. Apart from communicating with the actuator module, it should be able to connect to various devices/sensors that we might need - e.g. a UART port for live debugging between the hardware and our computer. If we need to satisfy these requirements, the most obvious concept is a development board with a micro controller, GPIO pins and power supply. Below is a list of some of the *many* boards that satisfy these needs:

- Arduino Boards
- ESP32
- STM32
- Teensy
- (OpenRB-150)

These development boards all more or less satisfy our needs, and the factors of availability and cost end up determining our choice. The fact that we have multiple Arduino UNOs available already and that we are used to developing on these, means that we will use the Arduino UNO for the control hardware. (Note: the OpenRB-150 is very well suited for controlling the Dynamixel motors, but would increase the cost of the project. For a future improvement of controller size and performance, using the OpenRB-150 micro controller would be the obvious choice).

As mentioned in section 4.6, the Dynamixel motors use a half-duplex serial connection. Since the Arduino UNO uses a full-duplex UART, this means that apart from the Arduino itself we need to use a full-to-half-duplex interface between motors and micro controller - which exists in the form of a shield for the Arduino UNO.

The last addition to the micro controller itself, is a separate UART to USB debugging tool. We need this separate UART since the UART connected to the USB port of the Arduino UNO is used for the Dynamixel bus. So for simultaneous use of the debugging port and motor communication we need the extra UART to USB.

Signal pathway

The Dynamixel motors communicate using the Dynamixel Protocol 2.0 (via the UART TLL Half-Duplex connection) [24]. This protocol is the backbone of communicating with the motors, and enables us to set up the motors, command them and retrieve information about them. The protocol relies on the instruction packet structure and response (status) packet structure seen in tables 4 and 5, respectively.

Header	ID	Length	Instr	Param	CRC
4 bytes	1 bytes	2 bytes	1 bytes	Variable	2 bytes

Table 4: Principal structure of a Dynamixel Protocol 2.0 packet [24]

Header	ID	Length	Instr	Error	Param	CRC
4 bytes	1 bytes	2 bytes	1 bytes	1 bytes	Variable	2 bytes

Table 5: Principal structure of a Dynamixel Protocol 2.0 status package [24]

To ensure fast responsiveness in the control and reliable real-time testing data acquisition, we want to limit the amount of time spent communicating on the Dynamixel bus. Therefore we will take a look at how the Dynamixel

bus works. As can be seen in table 4 the total size of the instruction packet is $10 + N_{\text{instr}}$ bytes, where N_{instr} depends on the instruction sent, since the parameters will vary from instruction to instruction. Likewise, the total length of the status packet (the response) is $11 + N_{\text{statues}}$ bytes.

The transfer time of requesting and reading motor data depends on the speed of transfer as well as the size of the data. The Dynamixel motors support up 4 Mbps of data transfer rates, however, the Arduino UNO does not support such high data rates. During initial research and prototyping, we found that a transfer rate of 1 Mbps balanced speed and stability. Hence, we have chosen a transfer rate of 1 Mbps. Since the Dynamixel Protocol UART uses 8-bit, 1 stop/start bit, no parity setup, the effective UART transfer speed (r_{eff}) of data in bytes/second can be calculated like this:

$$r_{\text{eff}} = \frac{\text{bits/s}}{\text{bits/byte}} = \frac{1 * 10^6 \text{bits/s}}{(8 + 1 + 1)\ \text{bits/byte}} = 100,000\ \text{bytes/s} \qquad (1)$$

So when transferring data we end up with a transfer rate of $100,000$ bytes/s. If we now look at the sequence of transfer in the Dynamixel Protocol 2.0 (see figure 29), we see that the total transmission time includes a small delay between request and response (Return Delay Time, RTD), which is implemented by the motors.

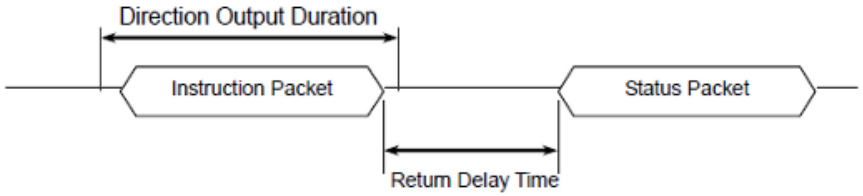


Figure 29: Time frame showing the different things affecting Dynamixel Protocol communication [24] (screenshot)

The purpose of RDT, is to avoid data corruption of the status packet (response), by avoiding bus collision. By default, the RDT is set to 500μs in the Dynamixel motor registers. To avoid messing with the timing on the bus, we keep the RDT unchanged. However, if higher transfer speeds were needed, the RDT value could be altered in the future.

Firmware design

The next concept to deal with in terms of controlling the gripper, is the actual code running on the micro controller and commands to the motors. We will now take a look at this part of the project.

To ensure smooth development and operation of the gripper, unity between program state and actual (physical) gripper state is important. Among other things, this includes accuracy and speed - which are linked together. A fast sampling of the physical state helps to achieve a high accuracy of the program state. Fast state sampling also enables the possibility to filter the current state, with less of a delay. To ease the coding and debugging of the gripper, we should ideally implement a state machine, that keeps track of the internal state and state transitions.

To solve the issue of bridging code running on a micro controller to real life state, robots are usually kinematically modeled. In soft robotics, the kinematic modeling is different from rigid robotics. Webster et al. [25] review the current techniques of modeling continuum robot arms using constant curvature kinematics, which might be of interest if we needed a model. Another example of correlating robot state and control is found in SpiRobs, where the effects of frictions from tendons are modeled [4]. In the scope of this project, we will not create a virtual inverse kinematic model of the gripper, but instead see how good of a performance we can achieve without any modeling, due to requirement of having a low control complexity.

On top of keeping track of internal state, grippers ideally need to be used and controlled by other robotic systems. This is usually done by designing an interface between the gripper's control system and the outside control. Many different communication protocols exists in the industry. Since this gripper would always be mounted on a robot arm, a wired interface makes the most sense. To name a few possibilities in our use case: CAN, SPI, I2C or RS-485 [26, 27]. However, because of the scope of the project, we will focus on a simple internal interface for now (button or timer). In a future version, the

addition of an industry standard interface protocol could be added, for better integration into a full robotic system.

Perhaps the most interesting part of the control subsystem is the actuation strategy implemented through the state machine. This will be the next thing we go through.

Actuation strategies

Actuation strategy is the way in which we actuate the tendon(s) in the gripper. This includes how many tendons we use, and the relative movement/force of the tendon(s) to each other. As mentioned in the conceptual design section of the finger (4.3), we most importantly need a flexing tendon, as this enables the grasping of object. There are two main reasons for adding an extensor tendon. If the passive extension of the fingers doesn't work, an extending tendon is also needed. And also, and maybe most importantly, as we learned in the beginning of the report, adding an antagonistic extending tendon can increase the stiffness of the structure.

In light of the control subsystem, there are three interesting aspects of the actuation curves: The type of control variable, the control signal curve and how that curve is sent to the actuator. By control curve we mean a control variable (e.g. PWM/position/current), that is changing over time - depicted as a curve. As mentioned in the beginning of the report, previous soft grippers use different strategies for grasping. In figure 30 we see some different types of control curves. The control variable (y-axis) can be things like motor voltage (PWM), motor current or motor position, depending on the gripper that is implemented. Having so many possibilities of choosing a control variable, resonates well with the discussion of having many operation modes available in the servos.

The choices we can make in terms of control curves are:

- Control variable
- Single vs. Dual tendon
- Symmetric vs Asymmetric
- Single vs Multi-stage

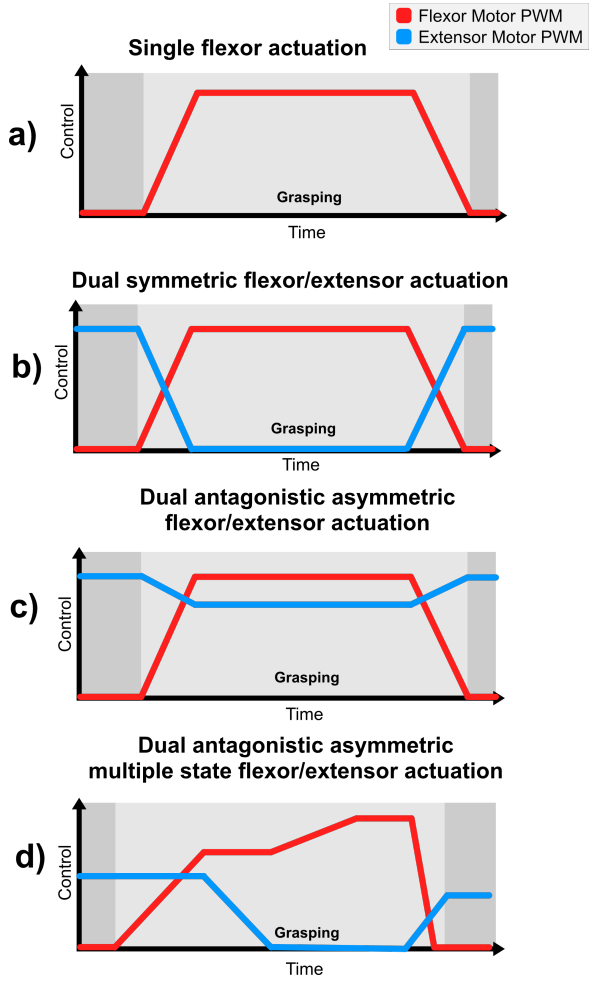


Figure 30: Sketches of conceptual actuation strategies. **a)** Simple single flexor actuation. Only the flexor is actuated here. **b)** Here both flexor and extensor are actuated in opposite, but symmetric curves. **c)** Again both flexor and extensor tendons are actuated, however, here the extensor applies an amount of tension against the flexing movement - resulting in an antagonistic movement. **c)** Both flexor and extensor tendons being actuated. The difference is that there are more stages than in the other strategies.

These differences balance implementation complexity and performance, and their strength depend on the final morphological implementation. Eg. if one is trying to achieve an alive/curious feel of the gripper like the one seen in SpiRobs [3], the fingers should be actuated using antagonistic torque and the advanced control curve seen in figure 30 **d)**, whereas if we just need a simple grip action [6, 8] we can use the position variable and control curve **a)** or **b)**. We will discuss the final choice in the control section of the embodied design (5.6).

The practical question of transmitting the control signals to the motors also depends on which control curve, and specifically which control variable, is chosen. The internal control in the Dynamixel motors allows for defining custom position and velocity trajectory parameters, which can be used to realize control curves like the ones in figure 30 **a)-d)**. Then it is just a question of sending position commands to the motors simultaneously at the right times and the Dynamixels will generate the curves - resulting in the desired movement. However, if we want to implement control using motor current or voltage (PWM), we need to construct these curves on the controller and transmit the correct control value at each instance in time.

This is the end of the control section, and thus also the end of the *conceptual* design as a whole. We have now laid out the concepts that are crucial to consider in the prototyping stage of the project: *embodied* design.

5 Embodied design

We are now ready to embody the conceptual designs from the previous section into actual physical structures. This is an iterative process, and through continuous refinements in each version, we converge on the final design of the gripper. This development process is described in the following.

5.1 Development of test platform

The first "prototype" we developed, was a small testing platform (figure 31), that enabled us to do some initial testing of both the actuation, control and finger structure. It was great for a quick start and sparking the intial ideas. This test setup was later replaced, when we started building the final prototype, since we weren't able to test the transmission and finger mount properly on the test platform.

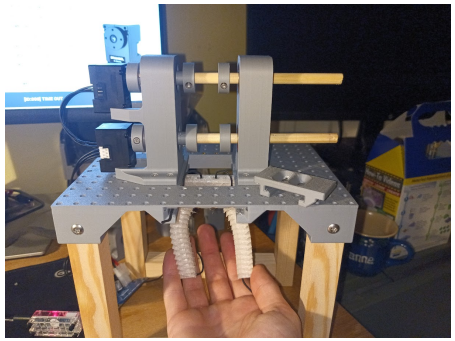


Figure 31: Test platform that we used for initial actuator and finger testing. The fingers can be slid in and quickly changed. The platform features a grid of small holes, where other objects can be fastened with M2.5 screws.

5.2 Fingers

As in the conceptual design, we start with the prototyping of the fingers. On the basis of the conceptual design we will in this subsection prototype and create a detailed design of the fingers for our gripper, with our human finger inspired principle function and octopus inspired morphology. The function-mean tree for the finger functions (Figure 14) gave us the relevant functions with different means to perform these function. In the following we want to detail these means, which leads to the making of several different prototypes.

Design and testing of joints

The flexure joints is the most important aspect, as it enables bending, and therefore we prototype the flexure joints first. We want our flexures to have low internal resistance and be stable. To find flexures living up to these requirements we prototype the flexures mentioned fig. 18 and they can be seen in fig. 32.

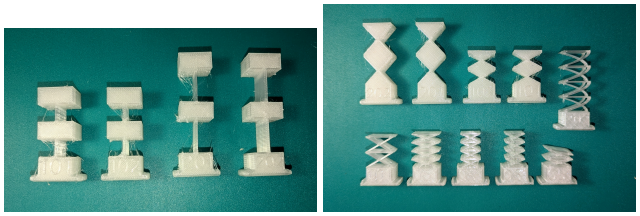


Figure 32: **Left:** The first iteration of blade flexure joints. **Right:** The first iteration of living hinge and cross flexure joints. Top row shows living hinges. Bottom row shows cross flexures.

After testing the flexures by actuating them by hand, we made these findings:

- **Living hinges:** plastically deformed when actuated too many times, since all the stress was concentrated at one point. The springiness of the flexure were extremely slow, this can be seen in fig. 34
- **Blade flexures:** worked well, and we decided to test them using the test setup in fig. 33 and the results in table 6 showed that shorter and thinner flexures were more stable and has less internal resistance.
- **Cross flexure:** In figure 32 we can see the cross flexures on the bottom row. The cross flexures showed best springiness out of the three, which can be attributed to the extra material in which the stress can

distribute inside the flexure. And we validated that they worked well by testing with the setup in fig. 33 giving us the results in table 7

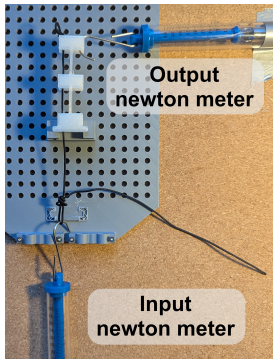


Figure 33: Blade flexure test setup for finding the effectiveness of the joints. The output newton meter is taped to the table to ensure, it is grounded and the only changing variable is the force applied with the input newton meter.

Length	Thickness	Input Force	Output Force	Notes
20 mm	2.0 mm	10 N	1.0 N	Stable
20 mm	0.8 mm	10 N	—	**Buckling**
10 mm	2.0 mm	10 N	1.3 N	Stable
10 mm	0.8 mm	10 N	1.6 N	Stable

Table 6: Blade flexure test results.

Length	Thickness	Input Force	Output Force	Cross Angle
5 mm	0.8 mm	10 N	1.2 N	32.5°
10 mm	0.8 mm	10 N	1.5 N	57.2°
5 mm	0.8 mm	10 N	1.3 N	61.3°

Table 7: Cross flexure test results.

We also did some initial tests of the springiness of the flexures (fig. 34, and found some problems:

1. TPU is a viscoelastic material. Material properties are time dependent, i.e. if a gripper is closed for a long time it would, take longer than normal to open again, this showed in our tests. This reduces the reliability of using the flexure joints as springs.

2. A flexure with more angular stiffness makes the flexure revert back to the original position faster and more reliably. It also makes the problem of viscoelasticity less obvious. More angular stiffness will however need a higher force to actuate the flexure.

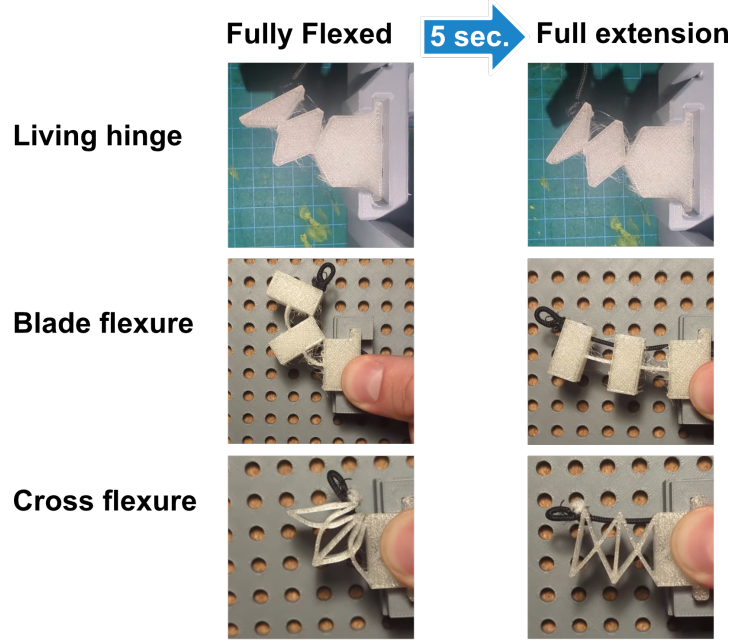


Figure 34: Testing the extension of the flexures, by fully loading the flexures, and checking the recovery 5 seconds after being loaded. The cross flexure showed the best extension capability.

To fix these problems we made a metal flexure finger (Appendix K) getting rid of these problems, but it did not get rid of the problems we will describe in the following.

First iteration gripper prototype

Using the results from above we made a gripper prototype utilizing the cross flexure for its superior extension, and for its ability to resist torsion (fig. 18). Using the test platform, and actuators we made some initial tests (fig. 35. The principle function from fig. 15 and principle function from fig. 16 worked great, but the choice of mimicking the morphology of the octopus to get a soft wrapping gripper did not work.

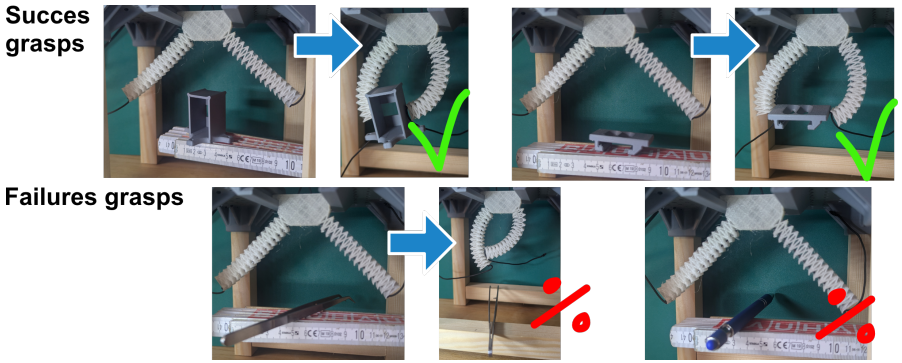


Figure 35: Test fingers of first gripper. Embodied version of the chosen concept in fig. 20

When flexing the gripper, it bended with an almost constant curvature, all the joints at once, causing no wrapping of the objects. As mentioned previously, an octopus grasps around an object by enveloping it by bending the arm joint by joint around the object to wrap it - following the logarithmic spiral. This together with the fact that this prototype did not address the balance between compliance and strength (fig. 11), made the grasping unreliable and only able to lift very light loads (maximum ca. 20 grams).

Dual actuation

With the findings from above in mind we prototyped tendons as a means to extend the fingers. Having two tendons also allows us to increase the stiffness by applying antagonistic forces. The mathematical model of a antagonistically tendon-driven robot by Wang et al. [4], tells us that by antagonistically actuating the finger, we can also actuate each individual joint one by one, due to the friction inside the tendon pulleys.

Our fingers with cross flexures buckle under these antagonistic forces (fig. 36). The FACT design methodology assumes infinitely stiff and thin flexures, i.e. the flexures are assumed to not buckle [16].

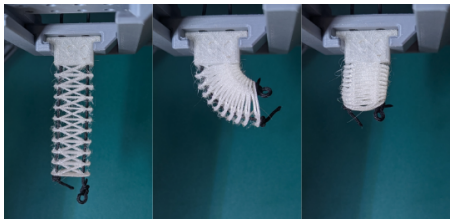


Figure 36: A finger with many cross flexures buckling under antagonistic actuation.

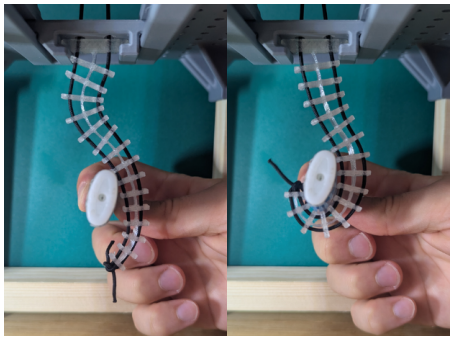


Figure 37: Using two tendons to control one finger.

Morphology

The example in fig. 37 appears to conform perfectly, however it is unreliable and does not always do that. To understand why we looked into the paper from Wang et al. [4], which modeled this kind of robot using the Capstan equation ($\frac{F_{in}}{F_{out}} = e^{\mu\theta}$). They found that the movement pattern of the robot is dependent on tension distribution inside the robot and its history.

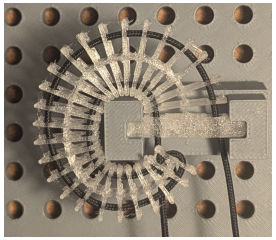


Figure 38: A fully extended finger with a non-spiral morphology. This finger would hit the finger mount, not being able to fully extend, and therefore act unpredictably.

When the finger in fig. 37 is fully extended, it collides with itself, creating an unpredictable next movement of the finger (as seen in fig. 38. To mitigate this problem we look at the octopus, and we find that the octopus curls it tentacles up and stores the tentacle in a logarithmic spiral, enabling them to uncurl their arms around an object.

We want our robot in a fully extended position to have the shape of a a logarithmic spiral. We therefore modeled our finger after the spiral as seen in figure 39. The spirals are generated from the general formula for a logarithmic spiral [28]:

$$\vec{s}(\theta) = ae^{b\theta}\begin{pmatrix}\cos\theta\\ \sin\theta\end{pmatrix} \tag{2}$$

The finger consists of a central spiral with parameters $a = 9 * 10^{-3}$ and $b = 0.2$. The inner/outer spirals (blue in the figure) are scaled versions of the central spiral ($k_{\text{scale}} = 0.5$).

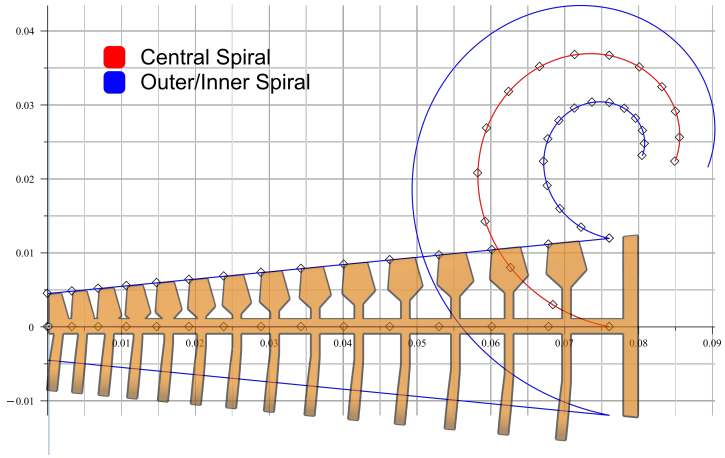


Figure 39: Depiction of the logarithmic spiral and points along its curve, spaced with equal increments in angle $\theta_{\text{step}} = 20°$. The spiral is "unfolded" and the points of the increments are used as a guidance for the finger joint design.

The points $((p_x, p_y))$ of the curled up spiral are determined with the formula for the arc length of a logarithmic spiral (starting in $\theta = 0$, which is determined by the line integral of formula 2), as well as the radial straight-line distance between two equal (angle) points ($\theta_{\text{central}} = \theta_{\text{outer}}$) on the central and outer spiral:

$$p_{\text{x}}(\theta) = \text{arc}(\theta) = \frac{a * \sqrt{1+b^2}}{b} * (e^{b\theta} - 1) \tag{3}$$

$$p_{\text{y}}(\theta) = |\vec{s}_{\text{outer}}(\theta) - \vec{s}_{\text{central}}(\theta)| \tag{4}$$

This point is then calculated 15 different times, with an increment of $\theta = 20°$ per point. Worth noting is that this point calculation does not preserve exact geometric thickness, but it maintains the exponential growth inherent to the logarithmic spiral. As a result, we can use the unfolded curve as an approximating template for the design of our finger. The final finger design can be seen in fig. 40 and the octopus inspired motion in fig. 41.

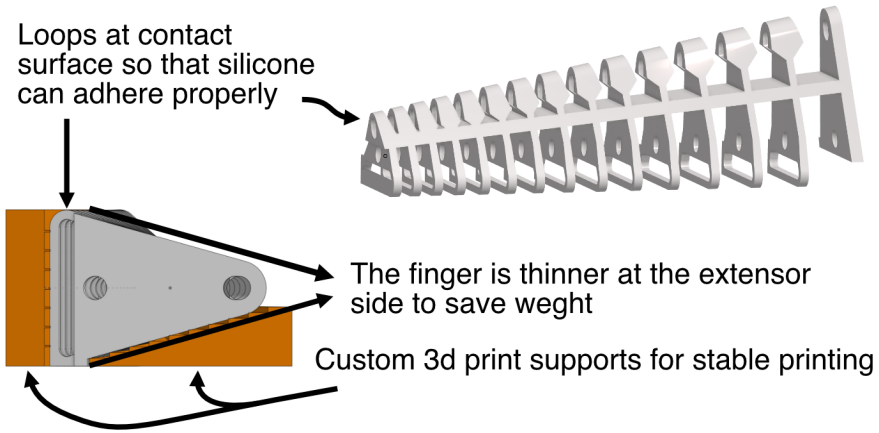


Figure 40: Final finger design. The custom supports enable better printing in TPU, while still saving weight from the tapered angle.

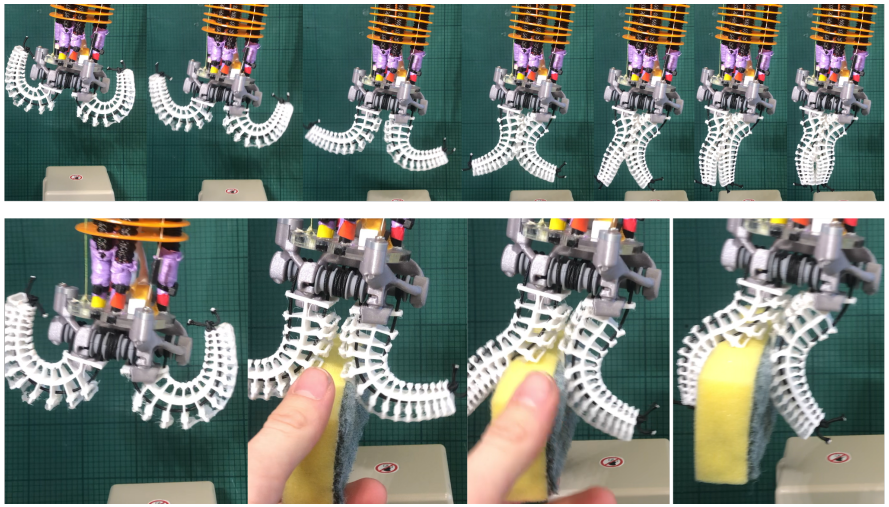


Figure 41: The octopus inspired wrapping behavior of the gripper

Design and testing of the contact surface

For the contact surface we prototyped both with and without silicone on the final finger and found that applying silicone dramatically increased load the load the gripper could carry. To make the silicone adhere properly, we created holes where the silicone would be, so the silicone did not have to rely solely on adhesion, but also on its own structural integrity. Applying the silicone to the gripper could be done in to ways. We could mold the silicone around the finger inside a form, creating a thick solid layer. This takes some time and needs tooling but is fairly robust. We could also dip coat the finger in silicone. This creates a thinner and more uneven layer, but is fast and does not require tooling.

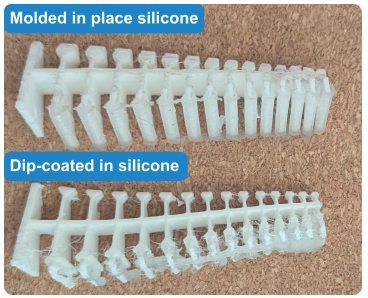


Figure 42: Different techniques to coat the finger in silicone. **Top:** Custom mold, **Bottom:** Dip coated silicone.

Fingertip

The current embodied design utilizes a manual knot to secure the tendons at the fingertip. While this approach minimized the complexity of the initial prototype, it introduced challenges regarding the calibration of the tendon lengths. Furthermore, the optimal geometry for the fingertip remains to be determined. As the current finger morphology is primarily designed for wrapping as seen in fig. 41 rather than pinching, the fingertip shape was not a priority in this the final design. However, future development should investigate fingertip geometries to enhance the gripper's ability to pinch.

5.3 Finger Mount

Now we will work on mounting these fingers to the arm. The concept chosen for our finger mount was a finger mount with drums for distributing the tendons. As we ended up with dual actuation in the last chapter, we choose the tendon distribution mechanism that enables dual actuation, shown here in fig. 43 with a vertical and horizontal drum, such that we had the possibility to use multiple fingers.

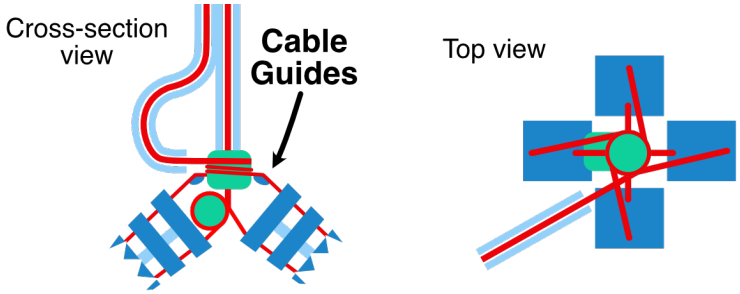


Figure 43: Chosen finger mount concept.

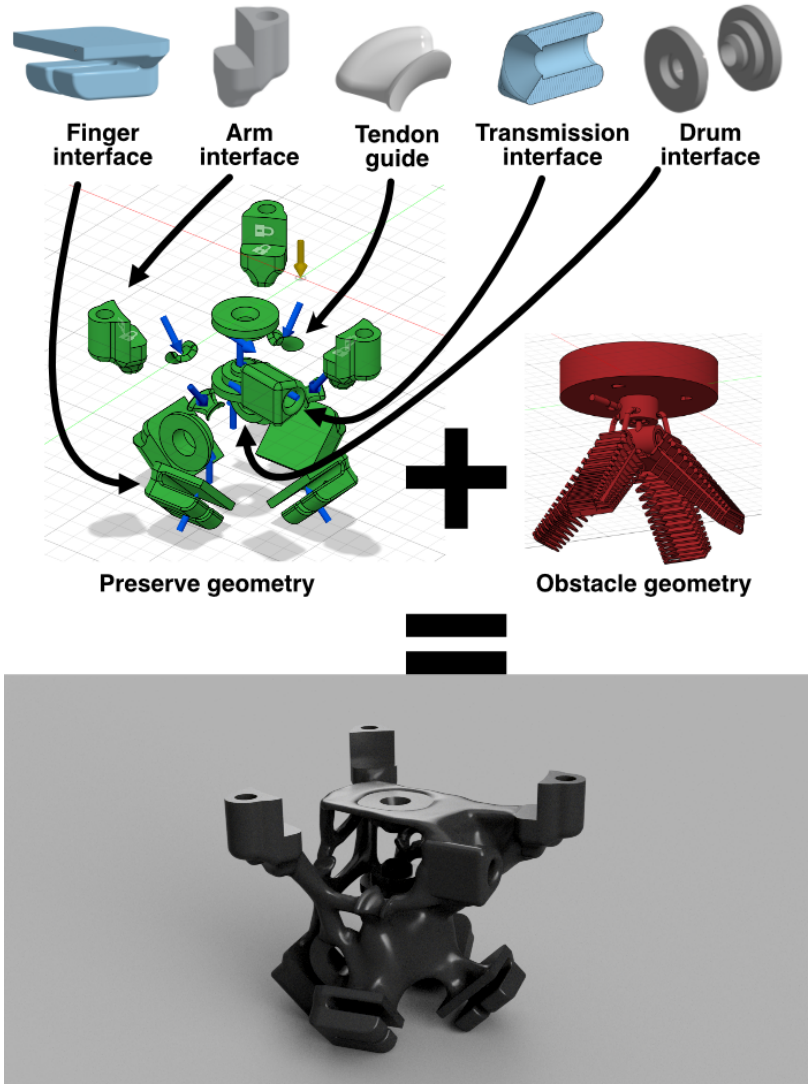


Figure 44: Generative design process of the finger mount.

Generative design and fabrication

We modeled the different functions (fig. 21) that the finger mount had in CAD, and then used fusion 360's generative design to connect them with lowest mass possible (fig. 44.

The resulting design was printed to house internal drums, however, this approach proved impractical. The structure had to be split into three separate components in CAD, printed individually, and then assembled around the drums. As we made several iterations of this fingermount fixing small problems, we found that this assembly was quite tedious, and one of the reasons we made a new prototype.

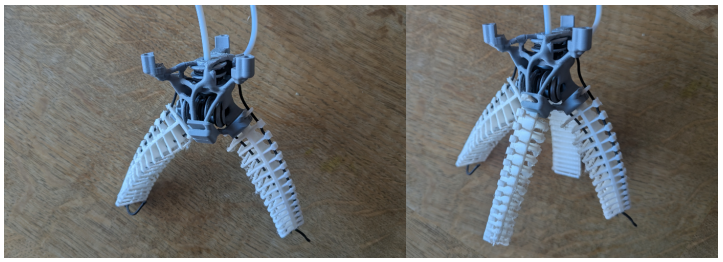


Figure 45: The first finger mount prototype with a vertical and horizontal drum. Equipped with two fingers on the left and four fingers on the right.

The gripper had low strength and was not able to lift items above 30 grams, not fulfilling our requirements from table 2. From how the fingers moved, we figured that the cable guide introduced too much resistance to the system. When the flexor tendon tried to flex, it also had to pull against the extensor tendon, which had additional resistance due to the tendon guide. We investigated this problem by measuring the resistance, and got the results in table 9. The test setup can be seen in appendix D. This validated our suspicion.

Type	Test count	Mean	Standard deviation	Margin of error	95 % Confidence Interval
Extensor	12	3.41 N	0.37 N	0.23 N	[3.17 N, 3.64 N]
Flexor	12	7.20 N	0.67 N	0.42 N	[6.77 N, 7.62 N]

Table 8: First prototype test results from testing how much force is output to the tendons when the finger mount is input 10 N of force. Details on the test can be found in appendix D.

The initial selection of the vertical drum configuration was driven by the desire to explore a multi-finger architecture. However, we decided to abandon the multi-finger capability and limit the design to two fingers based on the following rationale:

- **Grasp Stability:** As established in the conceptual design, each finger features a wide contact surface (20 mm). This increased surface area allows a two-finger configuration to resist torsional moments, providing a stable grasp without the need for additional fingers to fully constrain the object.
- **Weight Reduction:** Reducing the finger count lowers the overall mass of the end-effector.
- **Friction Elimination:** The vertical drum concept—which was necessary to enable radial multi-finger placement, was identified as the source of the excessive friction shown in Table 9.

Since there was no strict requirement for more than two fingers, and the "multi-finger ready" mechanism compromised performance, we pivoted to the concept with two horizontal drums, as this concept minimizes friction (Fig. 46).

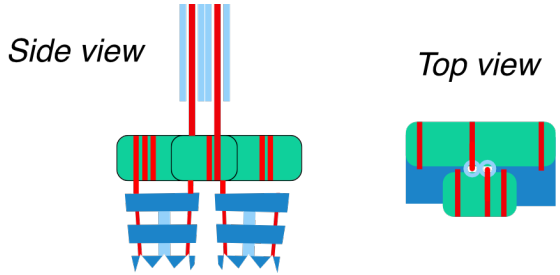


Figure 46: Chosen concept for second iteration

Second iteration finger mount

We went through the same development process as in fig. 44 (this time without needing the tendon guides, ending up with the design in fig. 47). We used miniature lightweight ball bearings, as we thought it would limit the energy loss as seen for the flexor in table 9, and it eased the assembly process, not having to glue together three parts. The assembly of this can be seen in appendix E.

We did the same test for the finger mount (results in table 9), and we found that the resistance from the tendon guide was statistically significantly reduced (table 10). Another thing to notice is that the flexor energy loss in the

finger mount did not improve, despite using bearings. Further development of the gripper could investigate this, and maybe get rid of the bearings to save weight.

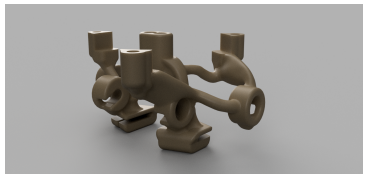


Figure 47: Render of the generative design of the second finger mount.

Type	Test count	Mean	Standard deviation	Margin of error	Confidence Interval
Extensor	12	6.05 N	0.55 N	0.35 N	[5.70 N, 6.40 N]
Flexor	12	6.85 N	0.48 N	0.31 N	[6.54 N, 7.16 N]

Table 9: Second prototype test results from testing how much tendon force is output to the fingers when the finger mount is input 10 N of force. Details on the test can be found in appendix D.

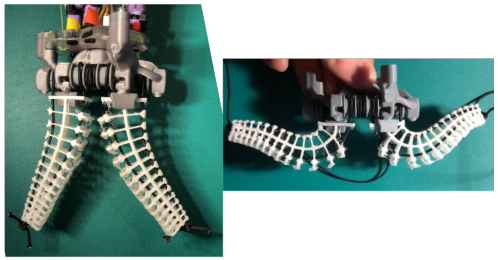


Figure 48: Second prototype of finger mount with two horizontal drums.

Type	Difference in means	Standard error	Confidence Interval
Extensor diff.	-2.64 N	0.19 N	[-3.04 N, -2.24 N]
Flexor diff.	0.35 N	0.24 N	[-0.15 N, 0.84 N]

Table 10: Independent difference of means analysis. Details on the test can be found in appendix D. There is a statistically significant lower energy loss in the extensor for the second iteration, compared to the first iteration.

5.4 Transmission

Section 4.5 describes how friction is the most important factor, and now we will test if the design choice of UHMWPE wire and PTFE tube, were in fact good choices. We will test this by finding the friction coefficient using the Capstan equation, with the setup in fig. 49. Isolating μ in the capstan equation gives us:

$$\mu = \frac{\ln(\frac{F_{in}}{F_{out}})}{\theta} \tag{5}$$

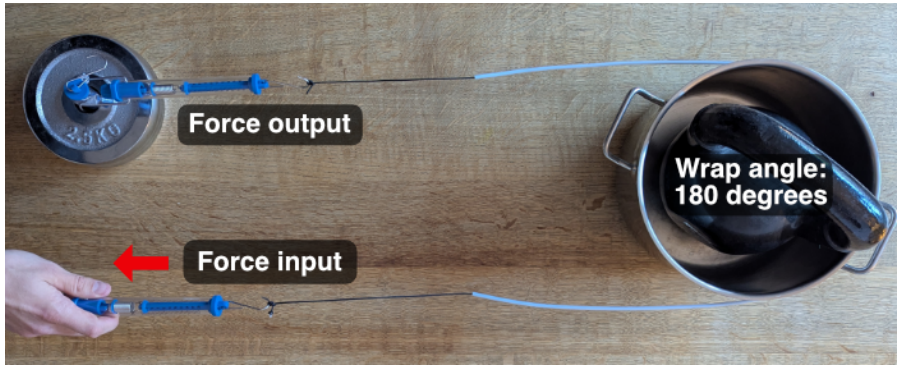


Figure 49: Transmission test setup. A 20 kg dumbell was secured inside pot, and one dumbell on one newton meter (F_{out}). The other newton meter (F_{in}) were pulled by hand, and the the results are shown in fig 50

We will then make a test setup as in fig. 49, using a wrap angle of $\theta = \pi$,

to calculate the static friction of the UHMWPE wire against the PTFE tube. The results are visualized in fig. 50. For the lower input forces we see higher static friction, up to 0.24. This could be due to som minor friction when the wire enters or exit the PTFE tube. As the wire gets tensioned, the PTFE tube will stretch more along the wire. However, we will not spend time validating this hypothesis, as the this difference in force is a minor detail. The important thing is that the static friction is pretty low at around 0.1. These results match the numbers from Matweb [18] of extruded PTFE against polished steel (0.02-0.2). We can hereby validate that UHMWPE wire inside PTFE tubing was a good design choice.

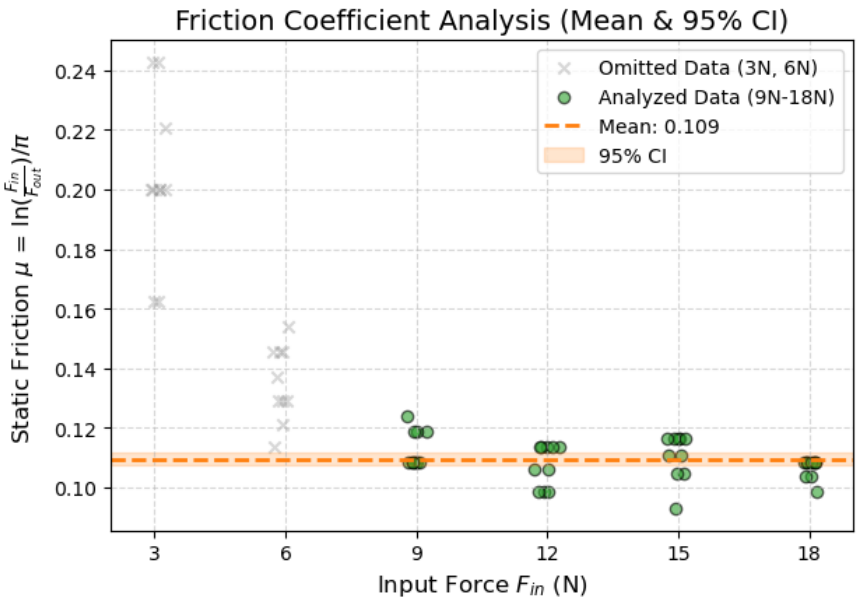


Figure 50: Using the test setup in fig. 49, 10 data points for 3, 6, 9, 12, 15 and 18 newtons of input force were recorded.

5.5 Actuator

We will now give a brief walkthrough of the process of prototyping the actuator subsystem. As we learned in the conceptual design (section 4.6), the actuator consists of motors, and enclosure, and an interface to the tendons. The actuator module also needs power and control - the former is provided from a lab power supply and the latter will be mentioned in the next subsection 5.6.

The complete actuator module setup is seen in figure 51.

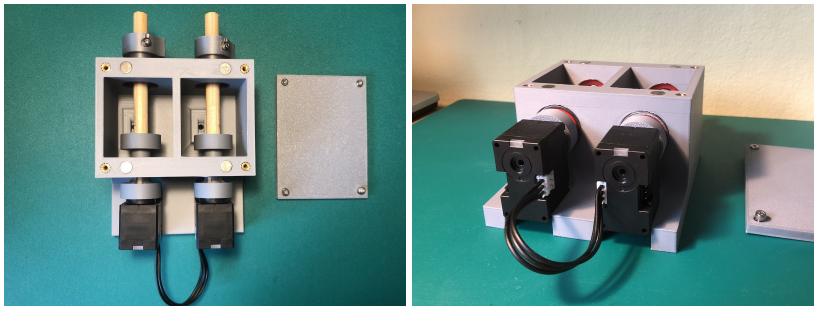


Figure 51: Pictures of the final version of the actuator module design. **Left:** Top view, looking into the enclosure where the axles are and the tendons are attached. **Right:** Side view, here we see the servo motors (XL330-M288-T) mounted on the axles.

Motors

In the beginning of the development phase, we experimented with both the large XL430 motors and the smaller XL330 motors. As mentioned in the conceptual design section, the main difference between these two motors are the weight, torque and ability to read and control electrical current.

We started out by using the big (XL430) servos in Position mode, but soon realized that we would like to experiment with the current based control modes of the smaller (XL330) servos. Therefore, we switched to using the smaller (XL330) servos early on in the prototyping process, so we could use both the Current-Based Position control mode and the Current control mode. These

two modes were a good stepping stone to the initial implementation of the desired compliant behavior.

During the design of the tendon distribution (finger mount), we ran into the problem of extremely high friction, which resulted in us looking towards increasing the actuation force. As mentioned, this could be achieved by using the bigger servo motors (XL430), but we would then loose the inherent compliant feel to the current based operation modes. Thus, to get an idea of how much improvement in performance we would achieve from switching to the bigger motors, we compared the two motors in a test. The results of this test can be seen in figure 52.

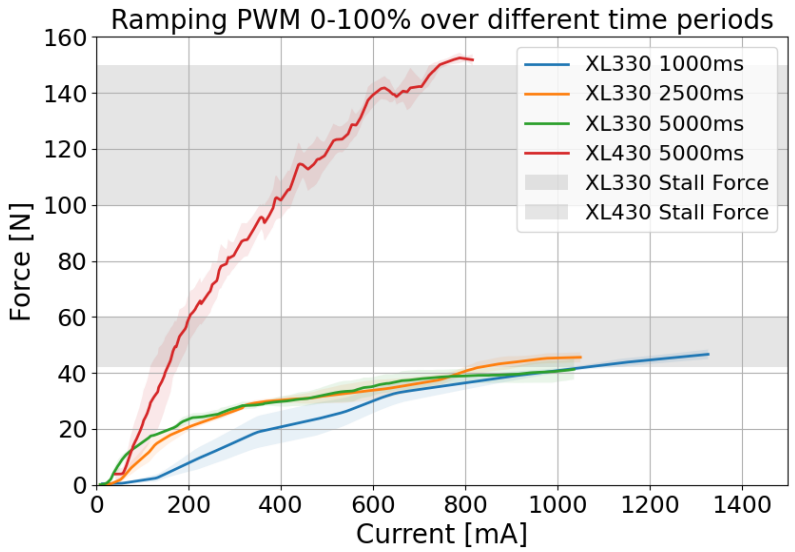


Figure 52: Measurements of force achieved from each motor, related to the (supply) current consumption. Results achieved by ramping PWM signal from 0-100% over 1000 ms, 2500 ms and 5000 ms, respectively. Force was estimated by pulling a load cell with a wire from a Ø8 mm axle + Ø2 mm wire wrapping. The test setup and procedure can be found in appendix A. Shaded areas are stall force rating from the motor datasheets.

As expected, an increase in motor current results in an increase in torque. This is due to the proportionality of these two measures ($\tau \propto I$) [29]. At a certain point, the motor torque flattens out, which is where we reach the the maximum achievable torque. From the bottom curves of the XL330, we see that the time period over which we ramp the motor PWM affects the curve, but not really the end result. From the test, the maximum achievable force from the small motor (XL330) in our constellation is around 40 N, whereas the bigger motor (XL430) can reach almost 150 N (datasheet values: 40-60 N and 100-150 N, table 3). Hence, we could increase the available torque by almost 300% by switching to the bigger motors. However, during the last design stages of the finger mount, we greatly reduced the amount of friction present in the system, which means that we did not actually need this increase in torque. This means that we landed on using the smaller motors in the final design. However, as will be explained in the embodied design of the control (section 5.6), we eventually switched back to using PWM as the control signal. This means that if the gripper actually needed more torque at some point, we could switch to the stronger motors (XL430), and re-tune the control.

Tendon Interface and Enclosure

The next part of the actuator subsystem, is the containment of the motors and the connecting of the tendons to the rotating axles in the motor. Before landing on the final version we went through a few other designs, the evolution of which are depicted in figure 53.

The first proof of concept was a small box that could be fastened to the test platform for initial testing. The shaft of the motors come with a pre-attached horn that enables us to attach any custom pulley horn-coupling with four small screws. This was first used to actuate the tendons and worked fine when the motor wasn't under high load. However, by observing the bending of the motor shaft, we quickly realized that the motors aren't made for large radial loads and hence needed a structure to support the radial load coming from the tendons. We chose to use bearings and wooden axles and made mounting points for these that could be fastened to the test platform. This worked brilliantly, and since the actuator is external to the gripper, the added weight and space is acceptable.

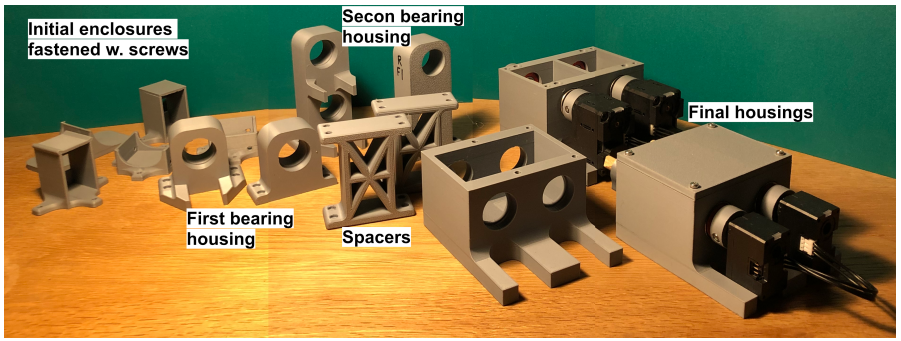


Figure 53: Evolution of actuator module enclosure

At a certain point we went from testing with a single finger to multiple fingers, meaning that we had to actuate eg. four or six tendons which (as mentioned in section 5.3) required some space for the tendon connections to move. This means that we had to raise the motor mounts slightly from the surface of the test platform. Therefore we added some spacers for the bearing house.

We ended up using axles, bearings, exactly constrained motors and a tendon guiding channel where the PFTE tubes attach in a standalone enclosure. The evolution of the actuator module can be seen in figure 53.

To summarize, the final version of the actuator module is composed of two Dynamixel XL330-M288-T servo motors, mounted onto a 3D printed enclosure that holds two separate 8mm wood axles (flexor/extensor) placed in standard (608 2RS) bearings. The tendons are attached to the axles with couplings and guided out of the enclosure and into the transmission PTFE tubes towards the fingers.

This marks the end of the walkthrough of the actuator prototyping process, and we will now move onto the aspect of developing the control of the actuators.

5.6 Control

The control subsystem is comprised of the code running on the micro controller (Arduino UNO + Dynamixel Shield). This includes everything from the external libraries, the state machine controlling the movements of the gripper, and the communication between motors and micro controller. As mentioned in the conceptual design (section 4.7), the reason that we chose this setup for the control part, is because of the easy setup, familiarity and availability of the parts. The physical setup for the firmware can be seen in figure 54.

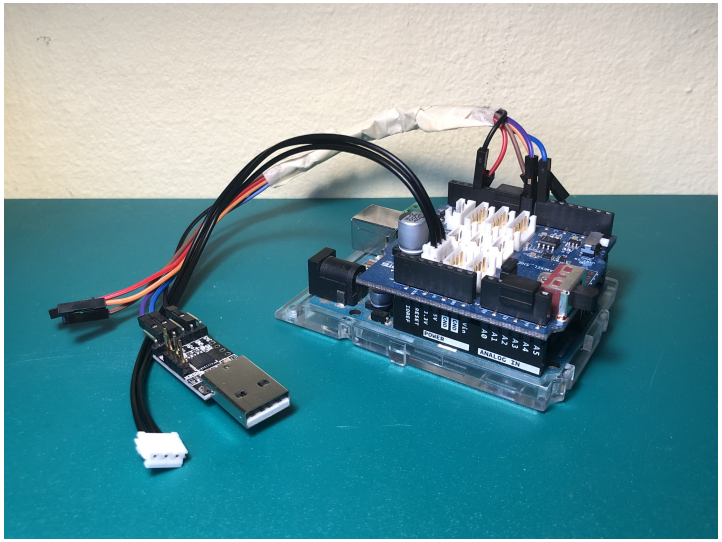


Figure 54: Arduino UNO with Dynamixel Shield attached. Serial debugger for simultaneous debugging and Dynamixel communication

Interface from micro controller to motors

To communicate with the actuator module, we need an interface between our microcontroller and the Dynamixel motors. As will be explained in the following section (4.7), we use an UART TTL Half-Duplex connection from the Arduino UNO to the motors. Since this connection is half-duplex, we need

an intermediate circuit on the bus, that handles the TX/RX switching. In our final design, this is achieved by using the Arduino Shield board, provided from Dynamixel. This shield has built in power supply connections, as well as a small selective buffer circuit which is controlled by a directional pin on the Arduino [30]. The matter of toggling this directional pin is handled by the Dynamixel Arduino Library.

Signal pathway

In our final setup we read the following four values from the Dynamixel motors to monitor the gripping process: `Present PWM` (2 bytes), `Present Current` (2 bytes), `Present Velocity` (4 bytes) and `Present Position` (4 bytes). These registers are all located in a sequential order in the Dynamixel's memory, meaning that we can read all four register values without making four subsequent calls. We must then read the first register, which is `Present PWM`, and 12 bytes from there. This reduces the amount of overhead in reading from the motors - thus reducing the active time on the bus. Even though this leads to a reduction in time on the bus, the biggest reduction, came from an initial increase of baud rate from 56400 bps to 1 Mbps - if another micro controller setup was used, we might go even faster.

Below is a calculation of the time it takes to request and read all four registers, once for each of the motors, which is done every time we update the state of the gripper. Remember that the instruction package size is $10 + N_{\text{instr}}$ bytes, and $10 + N_{\text{status}}$ bytes for the response (because of the extra error byte). Also recall that the effective transfer speed of data was earlier calculated to be $100,000$ bytes/s and the Return Delay Time is $500\mu s$. When sending a read instruction, we send the register address (2 bytes) + read length (2 bytes):

$$n_{\text{instr}} = (10 + 2 + 2) \text{ bytes/read} = 14 \text{ bytes/read} \quad (6)$$

$$n_{\text{status}} = (11 + 12) \text{ bytes/read} = 23 \text{ bytes/read} \quad (7)$$

$$t_{\text{motor}} = 2 * \frac{14 + 23 \text{ bytes/read}}{100,000 \text{ bytes/s}} + 2 * 500\mu\text{s/read} = 1.74 \text{ ms/read} \quad (8)$$

A read time of 1.74 ms is plenty fast in this scenario. Also, apart from the time spent reading motor state, the microcontroller also spends time writing the control variable to the motors every time period - we will not go through these timing calculations here. As will be explained in the next subsection, the actual sampling time in our program remains much slower, because of the max sample rate of the load cell ADC used. However, the motor sampling time of 1.74 ms still ensures that we run a tight control loop and makes it easier to speed up the loop of the code in the future.

Load cell

To perform tests that help us evaluate our gripper we wanted to use a load cell (see section 7). This meant that we needed to communicate with the motors while also reading a load cell value through an analog-to-digital converter (ADC). We have chosen a generic strain gauge sensor to measure forces, since they are cheap, easy to work with and readily available.

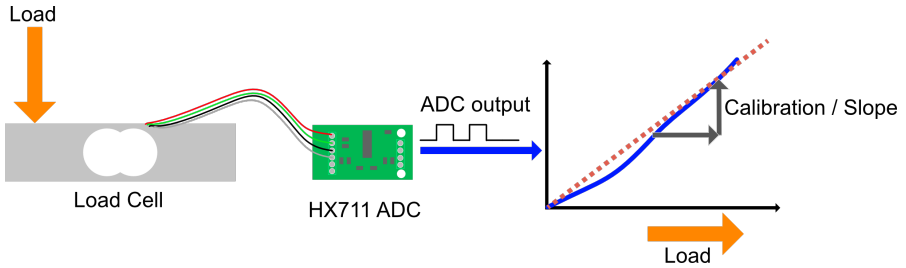


Figure 55: Diagram of load cell connected to the ADC (HX711). The load is estimated by detecting linear changes in the mV range from the load cell (Wheatstone bridge), and converting them to a digital ADC value.

To obtain usable data, we use the `HX711` ADC chip [31]. The HX711 is powered from the Arduino and has a `DATA` pin and a `CLK` line. The HX711 can be set to 10 samples per second (SPS) (100ms period) or 80 (SPS)(12.5ms period). The breakout boards for the HX711 are shipped as 10 SPS, but can be changed manually by de-soldering a zero ohm resistor. Changing the SPS from 10 to 80 will result in a faster conversion but a more noisy signal. During ADC conversion, the `DATA` pin is pulled high. Once the `DATA` pin goes low, data is ready to be read, which is done by pulsing the `CLK pin` 24

times. During our project we stayed at the 10 Hz sampling rate. Since the ADC conversion time is 100ms, we settled on a sampling frequency of 10 Hz, since this results in the lowest waiting time from the HX711 library, since the conversion is always ready immediately. Since the actual transmission of the converted data is negligible (normally $50\mu s$ [31]) compared to the conversion times of the HX711, we don't investigate it further.

Before we could use the load cell in our tests, we needed the calibration factor of the load cell. This calibration is done by measuring the ADC output at two different weights. The connection between load cell voltage (actually resistance) and load is assumed linear in the working range of the load cell (0-10 kg). The calibration factor is simply the slope of the line between the two measured points then [32].

Structure of code

Throughout this section we will go through the basic components of the code we used in the gripper. In general when writing any code, it is desirable to split the code base into different files for easier maintenance and usability. We have chosen to have the following different areas of the code:

- `Dynamixel Control`
- `Load Cell Control`
- `Serial Config`
- `General Config`
- `Main Control`

This gives us a clear division of functionality and responsibility in the code. The first two points (`Dynamixel` and `Load Cell`) are the functionalities that make the motors and the load cell work. The two next ones (`Serial` and `General Config`) are mostly for defining constants used throughout the program in one place. The last part is the `Main` file that uses code from the others and glues it all together in the right way.

Looking at figure 56 we see the flow of the program from the main file. From here it is really quite simple and consists of the initialization, a button detection snippet and a timed action, that reads values from the motors (and load cell if needed) and updates the state of the gripper depending on current state, motor values and button input.

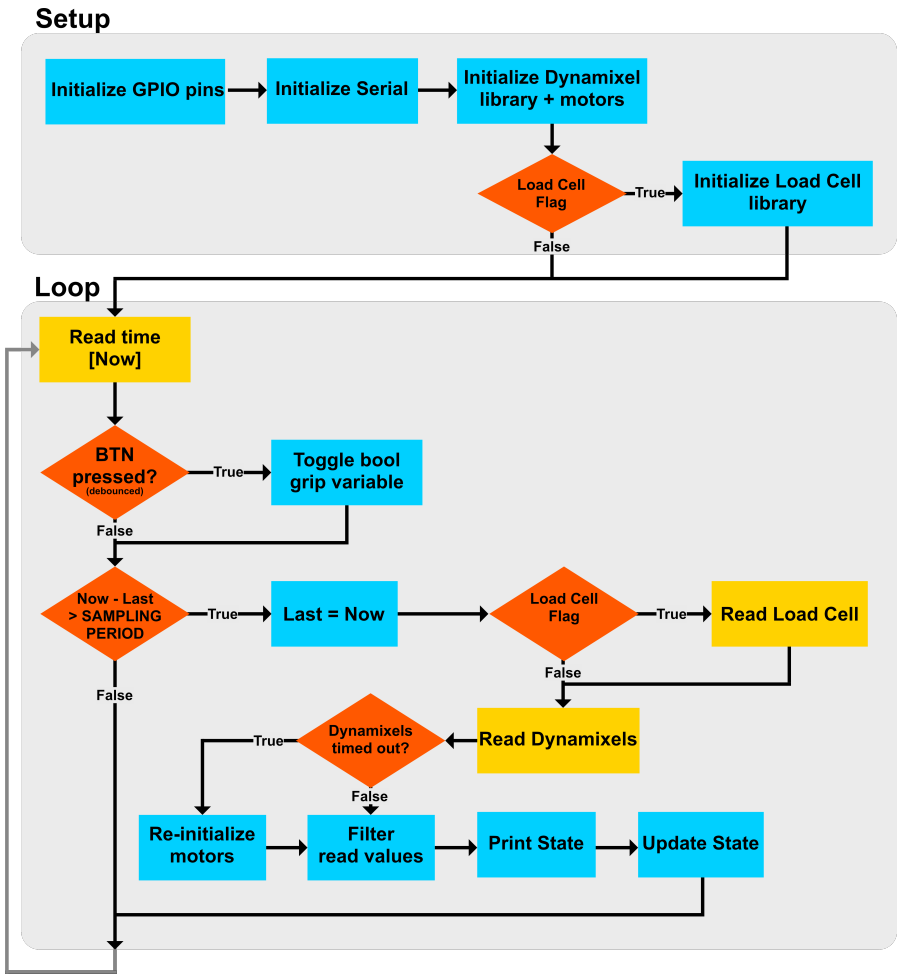


Figure 56: Flow diagram describing the final version of firmware controlling the gripper.

A remark worth making on the timing of the main part of the code is the choice of using the same timed action to read the load cells and read the Dynamixels motors. Since the ADC (HX711) reading the Load Cell values are run at 10 SPS, this loop will be limited to a `SAMPLING_PERIOD` of 100 ms whenever `Load Cell Flag` is true. As found earlier, the motors can be controlled way faster than this, and hence in a future version, it would make sense to separate the Load Cell sampling from the motor sampling and State updating. We did not do this yet because of three reasons:

- The 100 ms common timing did not worsen the gripping performance of the gripper substantially.
- The 100 ms sampling period seemed sufficient for our tests (see section 7)
- We tuned the gripper control intervals at the 100 ms sampling period.

In a future version, one could implement the control commands independently of the sampling period, to further separate different code functionalities.

The second interesting part of the code is the `Dynamixel Control`. This file contains all the logic for communicating with the motor, as well as the function that updates the state of the gripper. It handles the initialization of the motors (`dxl_init()`) where we open the communication and set the appropriate setting registers in the motors. Here we also ready the structure for the synchronous writing to the motors. When reading values from the motor, we use a function (`dxl_read_state()`) that reads all the wanted registers as a single feedback block and handles the parsing of this feedback block into separate state variables. For checking if the motors are still alive, we use the Dynamixel library's built in error code function in a custom wrapper function (`dxl_motors_timeout()`).

Finally, the state machine of the gripper is implemented (`dxl_update_state()`) and called from the main loop in every sampling period.

The functionality enabling other robotic components to communicate with the gripper is still very simple. As mentioned, we currently just use a button for toggling the gripping command. Technically, another robotic system could control the gripper like this (simple grip/don't grip) through a pin, but if we were to further develop the gripper, we would probably implement a proper protocol - as discussed in the conceptual design of the control. Furthermore, we would look into a "detection-algorithm" to enable feedback of a successful grasp to an external robotic system.

We will now move on to explaining the different control variables that we were able to control the motors with.

Operation modes

Both the big (XL430) and small (XL330) servos support different operation modes; meaning different ways of controlling them. Early in the prototyping, we used the big (XL430) servo in position mode. We observed how much the position of the motor moved when going from an open to a closed grip. Since the motors don't remember their absolute position out of the range of one revolution (360 degrees) between power cycles, we needed a manual calibration sequence to run at every power up of the gripper.

The position control worked okay as a starting point, but we soon thought that since the big (XL430) servo lacked a current based operation mode, it would be hard to get a simple compliant behavior out of them. Thus, we switched to the smaller motors (XL330), since they have both a current based control mode and a current based position control mode. This worked fine and we were even able to detect an object collision with the current based position control mode (see appendix L). Because the current based position control goes to a specific position but never exceeds a set limit current, this gripper motion was compliant enough.

At last, we wanted to experiment with a true antagonistic tendon actuation, meaning that we had to switch to either PWM mode or current mode, since these directly control the force of the motion. After experimenting with current mode, we found that we were now no longer able to detect collisions. This made us switch to PWM mode, since the current in current based mode never spiked (because it was controlled), and even though we still could not detect collisions, we landed on the PWM mode as a final control mode - also enabling us to switch to the big (XL430) servos later if the gripper needed more actuation force.

Actuation strategy

Apart from the control variable itself, it is also relevant to decide how this variable is applied. As discussed in the conceptual design section of the control, the relative force and movement of the extensor/flexor tendon is crucial for the outcome of the gripping movement. In the beginning we used an extremely simple actuation strategy, where the control curves where simply symmetric lines crossing each other. No antagonistic pulling other than the tension present from the pre-tensioning of the tendons. This strategy worked alright, but often resulted in a pinching grasp mode, and not a wrap. To try to change the grasping into favoring the wrap mode, we changed strategy.

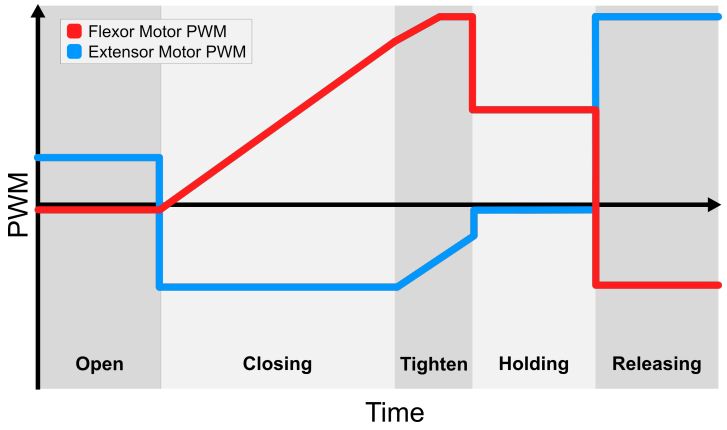


Figure 57: Diagram showing the PWM signal applied to the extensor/flexor motor in the different parts of the grasp cycle. The states (Open, Closing etc.) are toggled via the state machine. Note that a slight negative control signal results in a "light" holding back. So when the curves are close to zero, they actually still exert a slight antagonistic force.

The final strategy we landed on is loosely inspired by the strategy implemented by Wang et al. [3], with a multi stage gripping movement.

We have implemented a pre-tuned hard coded control curve (PWM multi stage signal), that in theory always performs the same movement pattern. By using an antagonistic unfolding pattern, where the extensor is "holding" back while the flexor slowly increases the pulling force, the finger aims to mimic an octopus unfolding its arm to meet and grasp an object. After the unfolding of the finger, we tighten *both* the extensor and flexor tendon, to achieve the characteristic antagonistically induced stiffness (note that in figure 57 a slight negative control signal results in a "light" holding back.). We then reduce the control signal significantly to overcome the issue of overheating the motors by applying a high PWM signal at motor stall. In the testing we have done, reducing the holding PWM works just as well as keeping a high PWM signal in the holding phase - the high PWM of the tighten phase overcomes the static friction in the actuation path.

When releasing (opening) the gripper, the extensor pulls with a high PWM, while the flexor releases the tension in the tendon. Going from an open gripper to a holding position takes around three seconds with the current tuning, but could probably be improved further.

This concludes the section on how we developed the control of the gripper, and thus also the embodied design as a whole.

6 Final design

Before moving onto the presentation of the evaluation and testing of the gripper, we want to make some comments on the final design of the gripper. This design can be seen in figure 58.

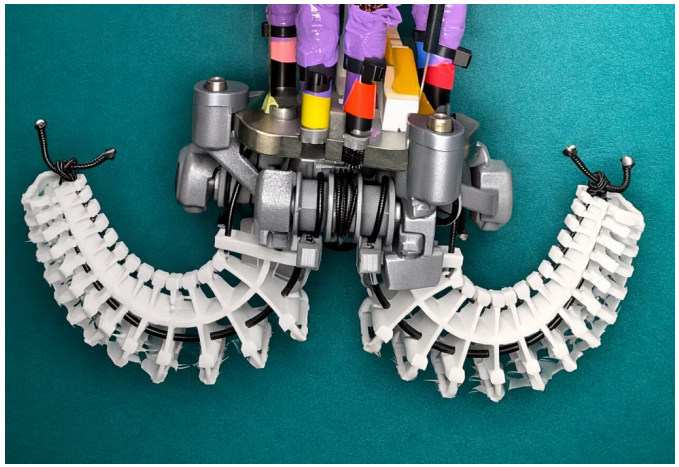


Figure 58: Final design of the gripper

The end-effector weight is 37 grams, which is below our 40 grams limit. The gripper is mountable on the NRT robot arm with three fastening points. Furthermore, the two PTFE transmission tubes are able to fit inside the robot arm. However, the transmission tubes might affect the movement of the robot arm, by pushing or pulling on the end-effector base. To mitigate this problem, the transmission tubes are placed on the NRT Lab arm so that they are able to move freely, when the arm moves.

The gripping speed of the gripper is consistently around three seconds which uphold our grasping speed criteria. We could probably make the grip even faster, with additional fine tuning for speed.

The control of the gripper consist of a multi-stage control curve, but is still relatively simply, by using direct PWM feedforward control. In terms of the assembling complexity, the manufacturing of the gripper remained acceptably simple, by being 3D printed and glued together without any special tools. The assembly guide for both the actuator module and gripper itself can be found in appendix E and appendix F.

Additionally, the bill of materials (BOM) can be found in appendix G. It is seen that if we only account for the actually used material, the final price of the gripper is approximately 105 euro.

The code used, the STL files for 3D print and the raw test data can be found in the public GitHub repository: `https://github.com/emilbroge/lightweight-soft-robotic-gripper-bsc-thesis`. Code has also been uploaded with the report as External Appendix I.

7 Testing and results

We now move onto the process of validating the performance of the gripper and evaluating whether it upholds the remaining requirements we set for the project in table 2. We will do this by performing two tests and go through the results of these in relation to our requirements. The purpose of the first test will be to estimate the maximum weight, where the gripper fails, and how that maximum weight correlates to the characteristics (size/shape) of the gripped object. The second test will be a simple "gripping success" test, that indicates how reliable the gripper is at gripping our initially defined test objects, while being mounted on the NRT Lab robot arm. It will thereby also test the grippers ability to hold soft objects.

7.1 Grasp modes and hypothesis

Throughout the development of the gripper, we have observed that the gripper has two overall grasp modes. We analyzed the two grasp modes in figure 59.

Pinching is where the gripper holds the objects from the side using only friction. This grasp mode is characterized by less surface contact area and the object can in some cases be held by only the tip of the fingers. This grasp mode relies on the grip force, and the friction coefficient between the gripper on the object surface. Hence, as long as the friction force is larger than the gravitational force, the object should be held in place (assuming no rotational moment on the object):

$$F_g \leq 2 \cdot \mu \cdot F_n \tag{9}$$

Wrapping is defined as when the gripper fingers are below the object, supporting the object. In this case, both friction and the support of the fingers underneath the object hold the object. We therefore hypothesize that wrapping can hold more load than pinching, if all other parameters are equal.

$$F_g \leq 2 \cdot \mu \cdot F_n + F_h \tag{10}$$

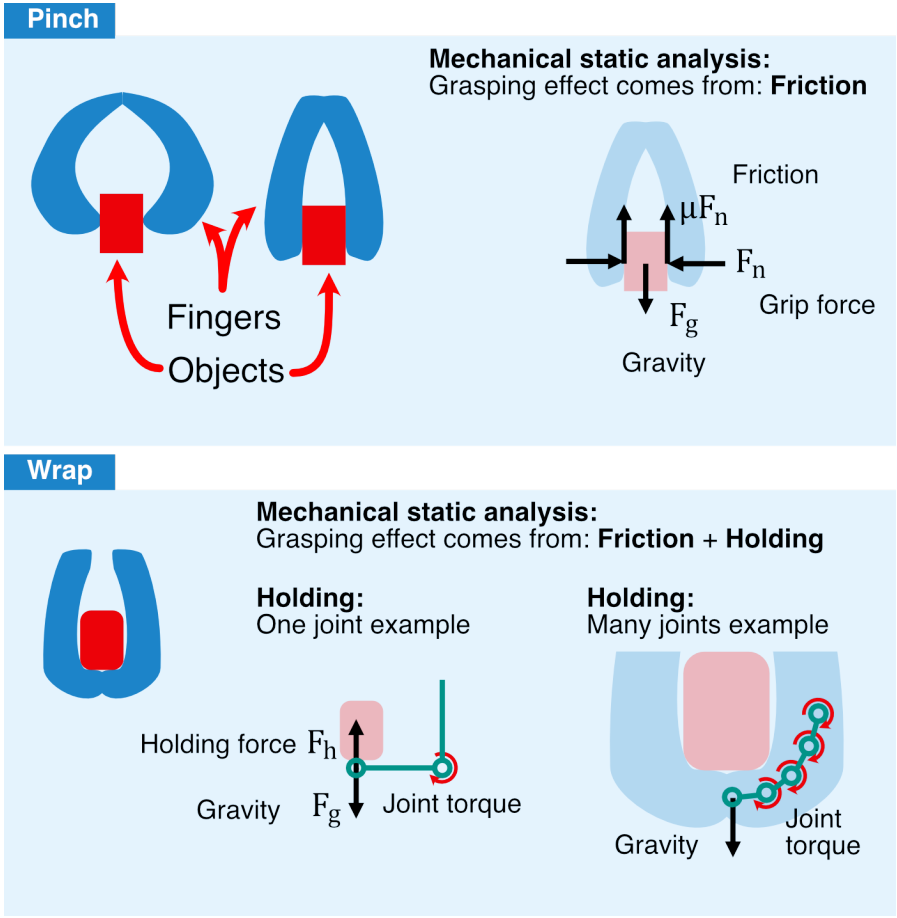


Figure 59: Grasp analysis for our gripper with the two grasp modes

7.2 Test objects

These two grasp modes can either be forced (from how you "hand" the object to the gripper) or can happen naturally depending on the object, the environment and the pose of the arm.

Before describing the tests, we make some initial comments on the grasps of the gripper. First and foremost, the gripper fingers only move in one plane, meaning that we can limit ourselves to vary test object parameters only in two dimensions, assuming they share the same center of gravity. Thus, to simplify the tests, we only vary the following: width and height of object + shape of the object. The shape will be kept to a box and a sphere. These variations are depicted in figure 60.

We want to investigate the gripping performance while varying the width, height and shape of the object, since these parameters will (perhaps) result in different outcomes, both in terms of grasping mode (wrap/pinch) and in terms of maximum carrying capacity. E.g. a very tall and narrow object will enforce the pinching mode.

Furthermore, since the maximum carrying capacity depends on the friction between the object and the fingers, we will keep the friction coefficient constant, by making all the test objects out of the same material (3D printed PLA+). Additionally we will test higher friction by printing test objects with "fuzzy skin", a print mode where the printhead *shakes* while printing creating a more rough surface

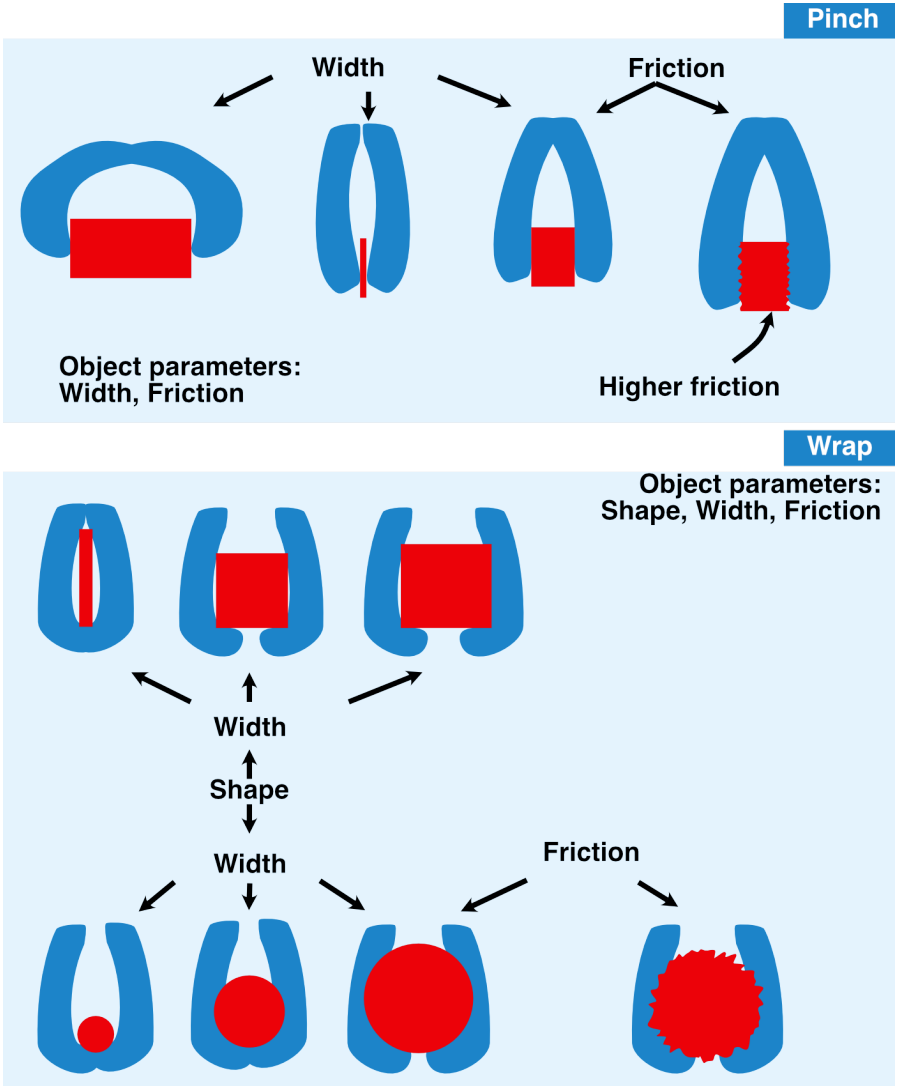


Figure 60: Diagram showing how we vary the parameters of the test objects. We can vary the width, height, shape and surface roughness.

7.3 Results: Maximum lift force

The first test that we conducted, was the payload "limit" of the gripper. This included using the custom 3D printed test objects, discussed in the last section. We attach these objects to a load cell with a rope and grip around the objects with the gripper. We then measure the lift force on the load cell (=weight of object) while increasing the lifting force on the gripper by approximately 1 N /s (using a newton meter), until the gripper no longer holds on to the object. We define the failure point of the gripper as the maximum reading, because that point in time is very prominent in the load cell readings and it resembles the maximum lifting capability. The test setup can be seen in figure 61 and a full description of the test procedure can be found in appendix B.

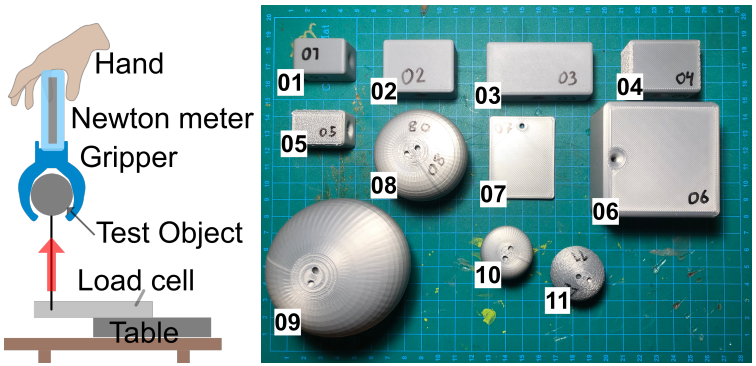


Figure 61: **Left:** Test setup of the payload/stability tests. The hand pulls the newton meter and gripper with an increasing force of 1 N/s, until the gripper no longer holds the object. **Right:** Overview of all the custom 3D printed test objects. 1-7 are all boxes and 8-11 are spheres.

The first plots we want to look at are the bar and box plot in figure 62. The bar plot gives an overview of the estimated payload capabilities of each individual test object. It can already here be seen that wrapping (green bars) generally enables a way higher lifting capability than pinching. The box plot at the bottom shows the distribution of the recorded means of the maximum lift achieved per object, where the two categories are wrap and pinch mode.

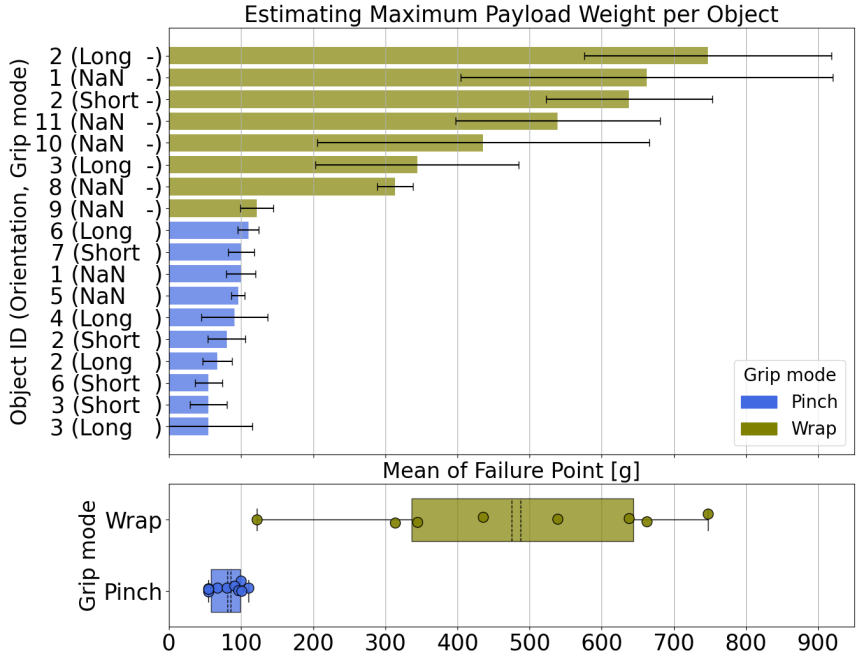


Figure 62: **Top:** Bar plot showing the mean and standard deviation of the failure points for each test object. The bars are colored depending on the observed grasp mode during the test. Long/Short side grasping is showed beside each bar. "NaN" means that the sides of the object are identical. **Bottom:** Box plot showing the distributions of the means of the pinching vs wrapped mode.

We again see a clear distinction between the two grasp modes, where the mean of the failure points in pinching is between 50-100 grams, and the failure points of the wrapping mode is between 125-750 grams. Apart from the higher mean in the wrapping box plot, the values are far more spread out than in the pinch plot. We believe that this higher variance stems from the many different ways in which the gripper can wrap around differently sized and shaped objects. So even if the grip mode is defined as a wrap for all the green bars, it might be tighter wrapped around one object compared to

another one. Furthermore, when the object is wrapped, we expect a combination of friction force and supporting forces to uphold the object. The balance between the friction force and supporting forces change when the wrapped object starts being pulled, and this results in a high variance in the measurements, depending on the object geometry.

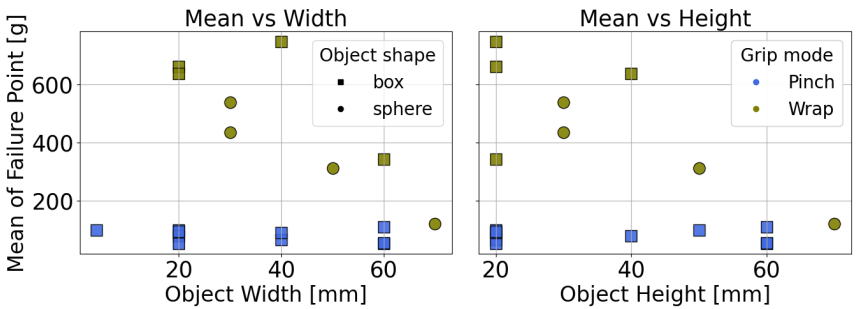


Figure 63: Plots of the mean failure loads for all the tests related to the grasp mode and object shape, width (left plot) and height (right plot).

We see that all the spheres "encourage" a wrapping grasp mode. The boxes include both wrapping and pinching, which makes sense, since we enforced both grasp modes.

We see that the pinching mode values almost always are the same amount of mean max payload, even when varying the height and width of the object. This makes sense, since it is only the friction between the gripper and object, that holds the object.

If we look at the wrapping mode, we see that a slight general trend of the mean max lift decreasing with an increase in object width and height. We believe that this is due to the geometry of the fingers. The more open the fingers are forced, the harder it is for them to properly wrap all the way around the objects, hence resulting in a weaker grip, because of the fact that the supporting forces from underneath are lowered - to the point where the grasp ends up as a pinch.

Test object number one and five are the exact same dimensions and shape, but number five has a rougher surface. From the results in figure 62, it is seen that the maximum payload of object one and five is identical. This indicates that increasing surface roughness without changing the underlying material (PLA) did not significantly alter the effective coefficient of friction in this test.

A little note on the usability of the results as a "maximum lift" test: The failure point is only the momentary maximum lift force that the gripper exerted on the test object. This is not necessarily the equivalent to the maximum stable payload capacity, but we still find the results from the tests useful. The test captured a pattern between the different grasp modes and the size and shape of the objects. Furthermore, we tested and observed the load cell output during the testing and noted that the maximum load cell output happened right before slipping conditions for pinching mode. For the wrap mode, the maximum load varied a bit more because of the dynamic "release" of the object. The gripper had to open up, before the object could be released, and this could be seen in the data. Translating this to a maximum payload would, however require more testing.

The measurements are calculated from the raw ADC load cell using the (assumed) linear relationship between resistance change in the load cell and weight (calibrated beforehand).

7.4 Results: Grip Success

The second test that we conducted was a testing of the grip success of the initial real world test objects (seen in figure 4). This test aimed to broadly evaluate the gripper real-life performance in grasping an object and holding onto it while moving the NRT Lab robot arm in a predefined trajectory.

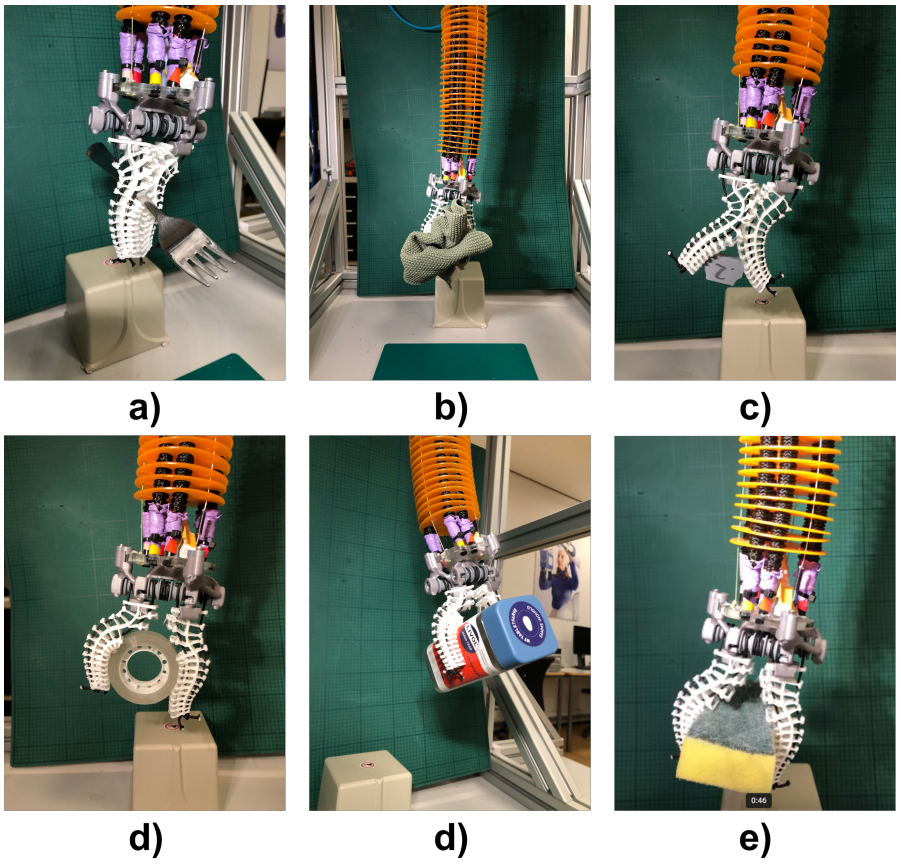


Figure 64: **a)** Fork **b)** Cloth **c)** "Lego" brick - pinch **d)** Tape, in between wrap and pinch **e)** Pill container, pinch **f)** Sponge, wrap

We therefore defined this test with a twofold outcome measurement; initial static grasping success and dynamic grasping success (when the robot arm was moving). Both measurements are either true or false, and are determined by observing if the test object is gripped successfully while the gripper is hanging at rest, and if the gripper successfully *holds on* to the test object while the NRT Lab robot arm moves in a predefined upwards spiraling trajectory, respectively. The full test setup can be seen in appendix C.

Since the results of this test is boolean, the test results are simpler compared to the previous payload test. If we take a look in figure 65, we see the success rates for the different objects. This rate is calculated from how many of the 10 trials we did for each object, that were successful.

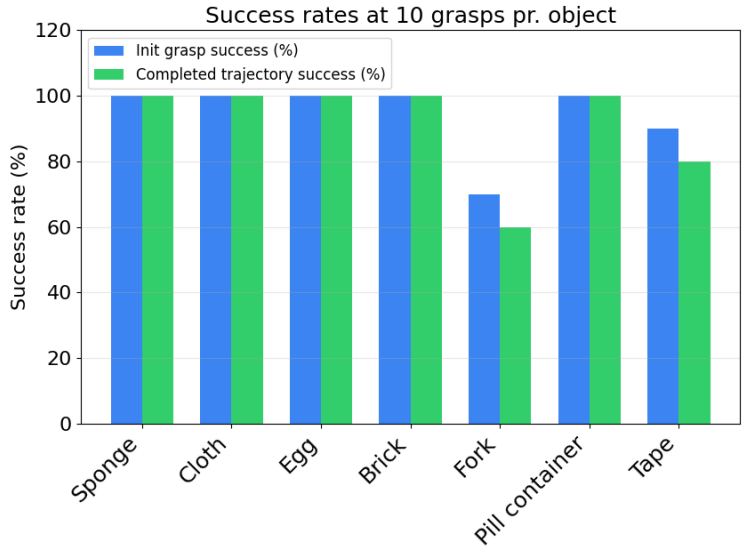


Figure 65: Bar plot showing the success rate of grasping the seven test objects and moving the gripper + arm in a predefined spiraling trajectory. Only the fork and tape had failure runs out of the 10 performed test runs.

Important to mention is the fact that the way the object was handed to the

gripper, varied slightly between rounds - simulating a more realistic situation. This means that for objects with multiple grasp orientations or placements (e.g. the tape or fork), the orientation was varied (equal amount of trials per orientation).

While doing the testing, we observed that this variation in how the object was handed to the gripper, affected the success rates. As can be seen in figure 65, five out of seven test objects had a successful grasp and movement rate of 100%. The fork and tape had non 100% success rates. For the fork the variation in success rate came from how close to the tips of the fingers the fork was placed - if we placed the fork all the way at the tip, it was harder for the gripper to hold onto the fork and resist its moment. For the tape, the variation came from the varying orientation of the object - when it wrapped around the slippery surface of the tape itself, it had a harder time holding on compared to gripping it from the side, where the gripper could wrap around inside the center hole of the tape.

8 Discussion

We have now reached the end of the report, and will discuss the process of design, developing and testing. We will also go through the different improvements that could be made to the final prototype in future work.

Process reflection

The process of design and development has been relatively structured and efficient. We have gone through the different concepts related to each of the identified subsystems, argued for the choices we made, and validated those choices. We went through several iterations on the design and think that an even better conceptual understanding could e.g. reduce the amount of iteration and physical prototypes made. But that is always easy to say that in hindsight.

If we had to change a thing about our approach to the project itself, it would be to ensure that the scope of the project was even better defined and realistically aligned, before starting the process of development. In the beginning we had high expectations for the project, and started designing with a big scope in mind. To ensure a smoother development, we could instead have focused on creating a minimal viable product first, and then expanded functionality from that. This would hopefully ensure that the design process stayed focused and purposeful throughout the whole project. That being said, the concept/embodiment and iterative design process *has* successfully guided us toward a working final prototype.

Future work

We have reached our goal of designing and developing a lightweight end-effector for the NRT Lab soft robot arm. Even though we are satisfied with this result, we will now go through some improvements that could be done.

The general improvement that could be done is developing every subsystem towards a more "finished product". With more time and resources the subsystems could be detailed and improved significantly.

Finger and finger mount design

The finger mount and fingers could be improved in terms of assembly complexity. The finger mount could sometimes snap during assembly, and we should therefore look into tougher materials than PLA, or fabrication methods without the layer adhesion weakness that comes from 3d printing. We could include a different tendon fastening mechanism, not relying on tying knots and manual length adjustments, when mounting the fingers to the finger mount.

The finger mount could be further improved by investigating if the small bearings actually reduces friction compared to having the drums mounted directly on the PLA finger mount structure. In terms of finger fabrication, alternative fabrication techniques could be investigated, which could improve the quality and level of detail of the fingers. As the finger design mainly focused on the wrapping movement, the fingertip design was not optimized. Future iterations should improve the fingertip to enable better pinching.

Miniaturization

In the future, the electronics and actuators could be redesigned and miniaturized. This could e.g. be in the form of using custom lightweight flexible

PCBs and switching to "pure" DC motors, with custom gearing and control. If done correctly, this could perhaps result in a feasible (in terms of weight) complete internal actuator design, that could be mounted on the arm itself. This would make the gripper more universal and reduce problem of routing the PTFE transmission tubes through the robot arm.

Control

The control subsystem could be improved by designing and implementing a better interface for other robotic systems to communicate with the gripper. Furthermore, since we in the beginning *were* able to detect a grasp using current sensing, we could investigate the possibilities for detecting a successful grasp, thus enabling extra feedback.

Durability

To improve the durability of the gripper, we should work on the surface on the fingers. We observed that the silicone had a tendency to loosen from the fingers, despite adding supporting structure for it. If the gripper is to be used continuously for many hours, this issue has to be addressed - the surface friction of the gripper is extremely important for successful operation.

8.1 Conclusion

This project demonstrated the design and development of a bio-inspired, lightweight soft robotic gripper through a structured conceptual–embodiment–testing framework. By abstracting biological principles, the gripper uses natural strategies for wrapping and compliance while being mechanical simple. The final design has a end-effecter weight of 37 g and is actuated by two externally placed Dynamixel XL330-M288-T servo motors driving artificial tendons made from UHMWPE wire, through ultra low friction PTFE tubing.

The fingers are inspired by logarithmic spirals observed in octopus tentacles. This design combined with a multi-stage antagonistic actuation curve, enables a favoring of the wrapping mode. Weight reduction and structural efficiency has been achieved through generative design of the finger mount. The system was 3D printed, with the fingers printed in TPU and the rigid finger mount printed in PLA, assembled using super glue and bearings.

Testing of the gripper shows a grip success rate of 100 % for 5 out of 7 test objects. The gripper's failure limits were identified, with pinch grasp mode failing between 50–100 g and wrap grasps between 125–750 g, showing the trade-off between compliance and load-bearing capacity. Overall, the project confirms that bio-inspired design combined with underactuated, antagonistic control can produce a highly lightweight yet capable soft robotic gripper.

References

[1] Sonal Keshwani, Torben Anker Lenau, Saeema Ahmed-Kristensen, and Amaresh Chakrabarti, ," *Iced 2013. International Conference on Engineering Design*, 2013.

[2] Torben A. Lenau and Akhlesh Lakhtakia, "Problem-driven biologically inspired design," *Biologically Inspired Design*, pp. 47–60, 2021.

[3] Zhanchi Wang, Nikolaos M. Freris, and Xi Wei, "Spirobs: Logarithmic spiral-shaped robots for versatile grasping across scales," *Device*, vol. 3, no. 4, pp. 100646, 2025.

[4] Zhanchi Wang and Nikolaos M. Freris, "Exploiting frictional effects to reproduce octopus-like reaching movements with a cable-driven spiral robot," *2024 Ieee 7th International Conference on Soft Robotics, Robosoft 2024*, pp. 537–542, 2024.

[5] Whitney Crooks, Gabrielle Vukasin, Maeve O'Sullivan, William Messner, and Chris Rogers, "Fin ray effect inspired soft robotic gripper: From the robosoft grand challenge toward optimization," *Frontiers in Robotics and AI*, vol. 3, 11 2016.

[6] Hareesh Godaba, Aqeel Sajad, Navin Patel, Kaspar Althoefer, and Ketao Zhang, "A two-fingered robot gripper with variable stiffness flexure hinges based on shape morphing," 2020, p. 8716 – 8721, All Open Access, Green Open Access.

[7] Fei Suo, Xiaolong Hui, Peixin Hua, Xuejian Bai, Jin Ma, Min Tan, and Yu Wang, "A biomimetic rigid-soft hybrid underwater gripper with compliance, stability, precise control, and high load capacity," *IEEE Transactions on Robotics*, vol. 41, pp. 3099–3112, 2025.

[8] Raymond R. Ma, Lael U. Odhner, and Aaron M. Dollar, "A modular, open-source 3d printed underactuated hand," *2013 Ieee International Conference on Robotics and Automation*, pp. 2737–2743, 2013.

[9] Jiaqi Zhu, Wenjie Fei, Han Ding, and Zhigang Wu, "Pressure and tendon actuation integrated three-finger soft gripper for wide force and speed range grasping," pp. 682–687, 2021.

[10] Renke Liu, Huakai Zheng, Maroš Hliboký, Hiroki Endo, Shuyao Zhang, Yusuke Baba, and Hideyuki Sawada, "Anatomically-inspired robotic finger with sma tendon actuation for enhanced biomimetic functionality," *Biomimetics*, vol. 9, no. 3, pp. 151, 2024.

[11] Haoran Li, Christopher J. Ford, Chenghua Lu, Yijiong Lin, Matteo Bianchi, Manuel G. Catalano, Efi Psomopoulou, and Nathan F. Lepora, "Tactile softhand-a: 3d-printed, tactile, highly-underactuated, anthropomorphic robot hand with an antagonistic tendon mechanism," 2025.

[12] David Bombara, Revanth Konda, Steven Swanbeck, and Jun Zhang, "Anthropomorphic twisted string-actuated soft robotic gripper with tendon-based stiffening," 2022.

[13] Richard G. Budynas and J. Keith Nisbett, *Shigley's Mechanical Engineering Design*, McGraw-Hill Education, New York, NY, 11th edition, 2020.

[14] Faulhaber, "Series 0816 ... sr," `hhttps://www.faulhaber.com/en/products/series/0816sr/` - last checked 04.01.2026.

[15] Mark R. Cutkosky, "A closer look at contact conditions," in *Robotic Grippers and Fine Manipulation*, chapter 5.4, p. 87. KLUWER ACADEMIC PUBLISHERS, 1985.

[16] Martin L. Culpepper., Massachusetts Institute of Technology. Dept. of Mechanical Engineering., Massachusetts Institute of Technology. Dept. of Mechanical Engineering., and Jonathan B. Hopkins, "Design of parallel flexure systems via freedom and constraint topologies (fact)," 2007.

[17] Robert Savage, "Fusion 360 introduction to generative design," 2019, Accessed: 2024-05-22.

[18] Matweb, "Overview of materials for polytetrafluoroethylene (ptfe), extruded," https://matweb.com/search/DataSheet.aspx?MatGUID=4e0b2e88eeba4aaeb18e8820f1444cdb - last checked 29.12.2025.

[19] AMCI, "Stepper vs servo," https://www.amci.com/industrial-automation-resources/plc-automation-tutorials/stepper-vs-servo/ - last checked 29.12.2025.

[20] Heidenhain, "Torque in servo motors," 2010, https://www.heidenhain.us/resources-and-news/torque-in-servo-motors/ - last checked 27.12.2025.

[21] Xinje, "What is analog servo? structure, principle and practical applications," 2025, https://xinje.com.my/uncategorized/analog-servo/ - last checked 28.12.2025.

[22] Dynamixel, "Dynamixel wizard 2.0," 2025, https://emanual.robotis.com/docs/en/software/dynamixel/dynamixel_wizard2/ - last checked 28.12.2025.

[23] Robotis, "Dynamixel u2d2," 2025, https://emanual.robotis.com/docs/en/parts/interface/u2d2/ - last checked 30.12.2025.

[24] Robotis, "Dynamixel protocol 2.0," 2025, https://emanual.robotis.com/docs/en/dxl/protocol2/ - last checked 10.12.2025.

[25] Robert J. WebsterIII and Bryan A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

[26] Matweb, "Robot communication protocols: A comprehensive guide," https://thinkrobotics.com/blogs/learn/robot-communication-protocols-a-comprehensive-guide - last checked 29.12.2025.

[27] Panav Arpit Raaj, "Communication protocols in robotics: Types and use cases," https://panav.gitbook.io/robotics-handbook/embedded-systems-for-robotics/communication/communication-protocols - last checked 29.12.2025.

[28] Eric W. Weisstein, ""logarithmic spiral." from mathworld–a wolfram resource," https://mathworld.wolfram.com/LogarithmicSpiral.html.

[29] Tutorials Point, "Torque in dc motor - armature torque and shaft torque," https://www.tutorialspoint.com/electrical_machines/torque_in_dc_motor_armature_shaft_torque.htm?utm_source=chatgpt.com - last checked 03.01.2026.

[30] Robotis, "Dynamixel shield," 2025, https://emanual.robotis.com/docs/en/parts/interface/dynamixel_shield/ - last checked 23.12.2025.

[31] Avia Semiconductor, "Hx711 datasheet," https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf - last checked 28.12.2025.

[32] Dan Schultz, "What are strain gauges?," 2025, https://www.dwyeromega.com/en-us/resources/strain-gages - last checked 31.12.2025.

Distribution of work

This section describes what areas we have each been responsible for. We divide the work into two main areas: design/development and writing the report. Even though we had our own areas of responsibilities, we generally worked together throughout the project, and included each other in processes and thoughts on the way.

Design/development

Mads

Bio-inspirational design, Mechanical design, CAD modeling, generative design in Fusion, buying materials, testing of prototypes, some coding

Emil

Electrical design, buying controllers and motors, testing of control, writing code, some CAD modeling

Both

3D printing, testing final prototypes, acquiring test data

Writing report

Mads

Conceptual design: Fingers and Bio-inspiration, Finger mount, Transmission.

Embodied design: Fingers and Bio-inspiration, Finger mount, Transmission

Emil

Conceptual design: Actuator, Control.

Embodied design: Actuator, Control.

Both

Introduction. Project overview & Design parameters. Previous Grippers. Testing and results. Discussion and Conclusion.

Part II

Appendix

Table of Contents

A Test: *Motor performance*

Purpose

We aim to characterize the performance of the two types of Dynamixel servo motors. Specifically, we want to test the correlation between current applied to the motor and output torque. This will be done by simultaneously measuring the current of the motor and pulling force on a load cell. Because of the proportionality between motor current and output torque ($\tau \propto I$), this test gives us a way to estimate (and compare) the forces applied at flex and extension (at a certain current)

Prerequisites

You need these things

1. Load cell (calibrated) incl. HX711 ADC Board
2. Arduino UNO
3. 5V power supply
4. Firmware for Arduino Uno
5. BetterSerialPlotter software
6. USB to UART TLL Converter
7. Dynamixel XL330-M288-T Servo motor
8. 8mm axle + Non-elastic wire (we use Dyneema 1.5mm wire)
9. Jig to mount Load Cell and Motor + Axle
10. Current sensor for measuring current in the big XL430 motor

Setup

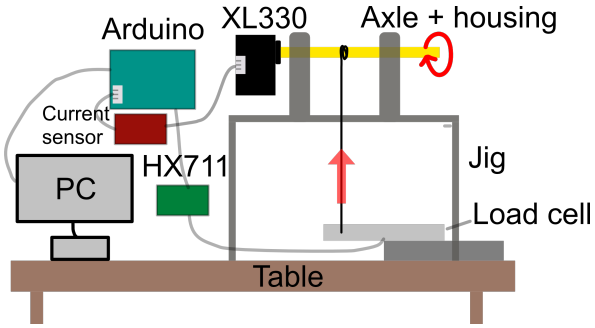


Figure 66: Full test setup, with side and top view.

Variables

During the test, these variables are changing. All other variables should be kept the same across tests.

- Current applied to motor

Quantitative measurements

You need to measure:

- The current applied to the motor
- The value from the load cell
- Radius of axle + wire

Sequence

1. Attach the load cell firmly (eg. to a table) and the motor to the axle + axle housing.
2. Connect the axle and load cell with a none elastic wire
3. While the elastic wire is not tight, tare the load cell. (Also measure the radius of the axle + wire).
4. Run the script that ramps current to the motors. Start collecting data in BetterSerialPlotter software. Repeat this step at least 10 times.

Results

You should now have a .csv file with Motor Current + PWM and Load Cell values. You should convert the load cell values to output torque of the motor with the following formula $\tau = w_{\text{lc}} * g * r_{\text{axle}}$ Take the average of the torque values in relation to current over the 10 repetitions and plot the points (current vs torque). Fit a linear curve ($\tau = k * I + \tau_0$) to the points. You can do the same with PWM against torque.

B Test: *Maximum Payload / Robustness*

Purpose

This test seeks to find the maximum payload the gripper robustly can hold before failure. The maximum payload depends on a lot of variables. Therefore, we use different test objects to test, under different conditions.

We attach a load cell to the objects to find a maximum payload, before the object is droped by the gripper.

Prerequisites

You need these things

1. Assembled gripper with electronics
2. Load cell (calibrated) incl. HX711 ADC board.
3. Test script
4. BetterSerialPlotter software
5. The 11 3D printed test objects:

Setup

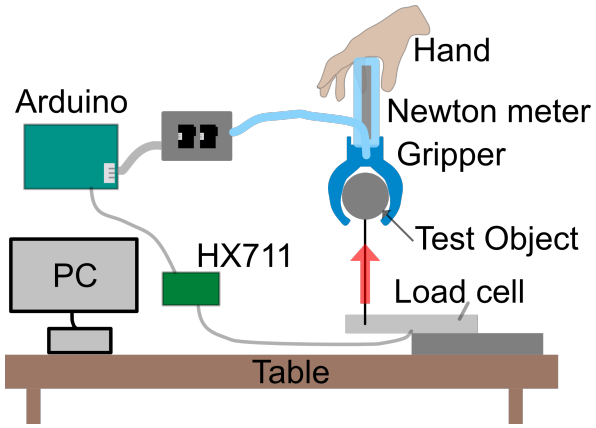


Figure 67: Full test setup

Variables

During the test, these variables are changing. All other variables should be kept the same across tests.

- Gripped object
- "Weigth" of object (lift force magnitude)

Quantitative measurements

You need to measure:

- Motor Current, PWM, Velocity, Position
- Load cell measurement

Sequence

1. Calibrate gripper and tare load cell
2. For each test object:
   (a) Assemble test setup with chosen test object. Note down the object.
   (b) Start recording data on the computer.
   (c) Initiate grip of object.
   (d) Lift gripper with the object gripped, until the rope tightens. (You should now not be able to move the gripper without applying force).
   (e) Increase the pulling force by 1 N/s, until the gripper opens or slips.
   (f) Repeat step 2c-2e at least 10 times
   (g) Save recorded data to a .csv

Results

For each object you now have a .csv file. You need to determine the lifting weight (from the load measurements) when the gripper failed. Take the maximum value of the load cell readings. Take the average of the force failures and save it for each object.

C Test: *Gripping Success Rates*

Purpose

Here we want to test the effectiveness of the gripper in gripping the specified test objects, while being mounted to the NRT Lab robot arm. We will both investigate a static grasp (arm is hanging still), and a dynamice grasp (arm is moving in a predefined upwards spiraling trajectory). We want to know how good the gripper is on average, meaning that each object will be gripped 10 times each

Prerequisites

You need these things

1. Assembled gripper
2. 5V Power supply
3. Firmware for Arduino Uno
4. Test objects (see exact definitions in requirements):
   (a) Egg
   (b) Lego brick
   (c) Cup
   (d) Tape
   (e) Cutlery
   (f) Cloth
   (g) Sponge

Setup

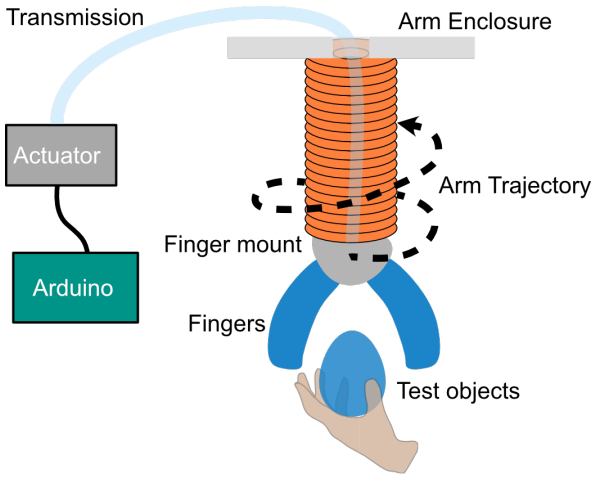


Figure 68: Full test setup

Variables

During the test, these variables are changing. All other variables should be kept the same across tests.

- Gripped object

Quantitative measurements

You need to measure:

- Successful initial grip? (True if: Gripper did not fail before started moving)
- Successful dynamic movement? (True if: The gripper held on to the object during the whole arm trajectory)

Sequence

1. Assemble test setup.
2. For each test object: (Note down the object)
   (a) Hand object to the open gripper, push button to close gripper.
   (b) Note down if initial grasp was succesfull or not.
   (c) Star the arm trajectory.
   (d) Note down if grasp was successful during the whole trajectory.
   (e) Repeat above steps 10 times.

Results

For each object you now have two sets of counts of successful grasps. Calculate the effectiveness of each object, both for initial static grasp and the dynamic grasping.

D Test: *Finger Mount*

Test setup and procedure

The test is setup like in fig. 69. The process went along like this:

1. First we ensured that the output force newton meter was grounded. If this step is omitted the results are not usable.
2. Ensure the finger mount is grounded properly, and attach the desired tendon output to the output newton meter. It is important that there is not slack in the newton meter. It is better that the tendon is a bit tight and pulling on the newton meter, as the newton meter can always be reset.
3. After resetting the output newton meter pull the input newton meter with 10 newtons of force and read the force output.
4. Write the reading down in raw data table, together with the test parameters, as done in table **??**

The analysis utilized an independent-samples t-test (also known as an unpaired t-test) to compare the means of the groups. This approach was selected because the observations across the different conditions were independent; specifically, each trial measurement in the 'Flexor old' group came from a different, unrelated specimen or subject than any measurement in the 'Flexor new' group.

Type	Count	Mean	STD	Margin of error	Confidence interval	
					Lower	Upper
Extensor old	12	3.41	0.37	0.23	3.17	3.64
Flexor old	12	7.20	0.67	0.42	6.77	7.62
Extensor new	12	6.05	0.55	0.35	5.70	6.40
Flexor new	12	6.85	0.48	0.31	6.54	7.16

Table 11: Summary statistics for force output across old and new tendon designs.

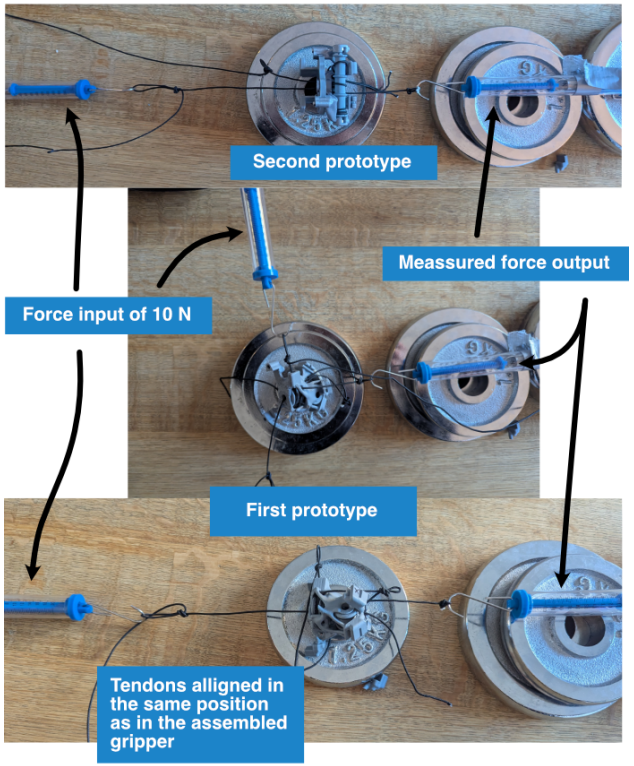
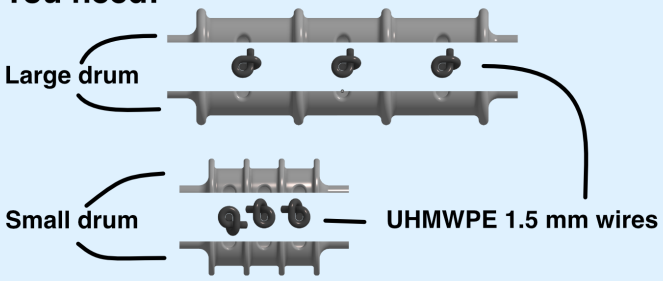


Figure 69: Finger mount test setup

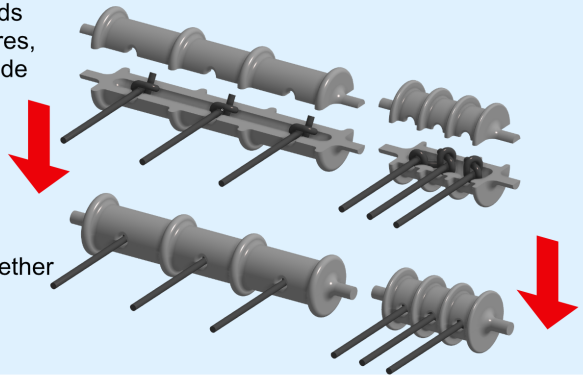
E Assembly: *End-effector*

Step 1: Drum subassembly

You need:



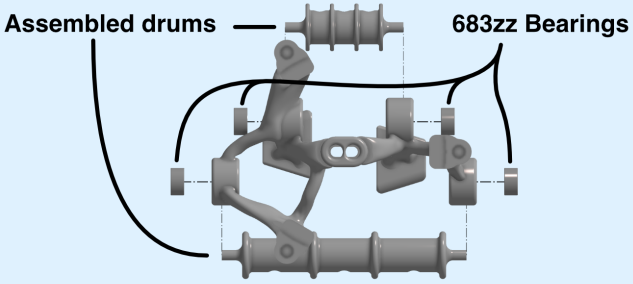
1.1 Tie knots on the ends of six UHMWPE wires, and place them inside



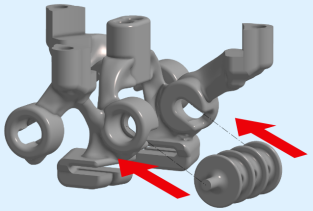
1.2 Glue the drums together

Step 2: Drum mounting

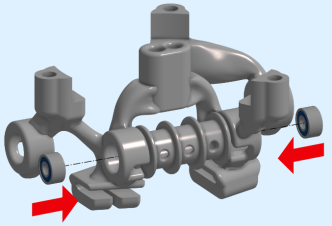
You need:



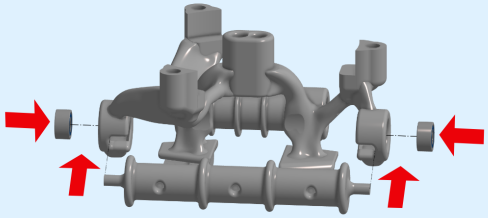
2.1 Insert the drum



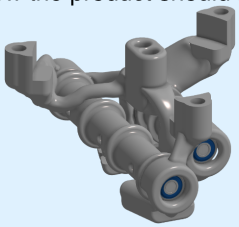
2.2 Push the bearings in on the drums. This is a hard press fit.



2.3 Insert the large drum and bearings



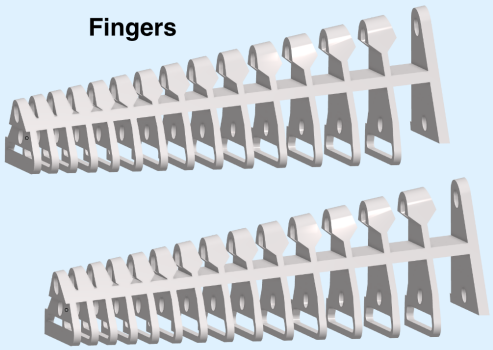
2.4 Now the product should look like this



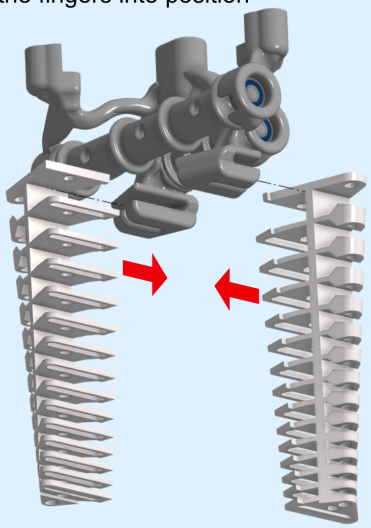
Step 3: Finger mounting

You need:

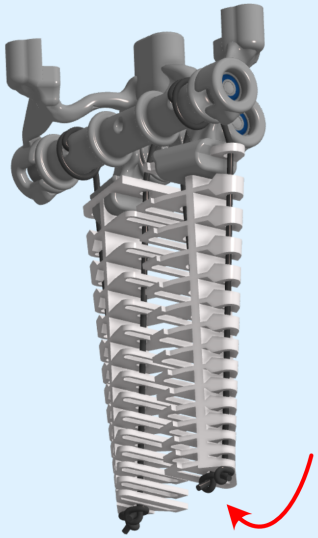
Fingers



3.1 Slide the fingers into position



3.2 Wind the short wires around the drum once, through the fingers, and tie a knot at the end.



Step 4: Arm mounting

You need:

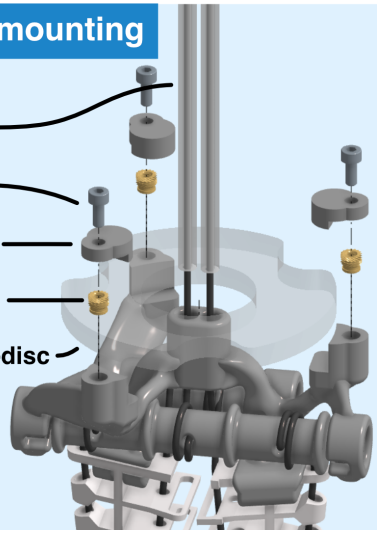
PTFE tubes

M2.5 bolts

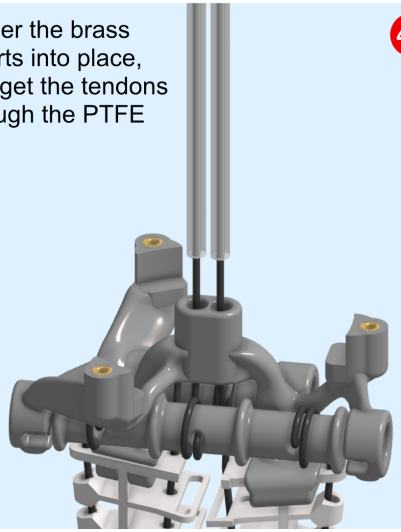
Arm holders

Brass inserts

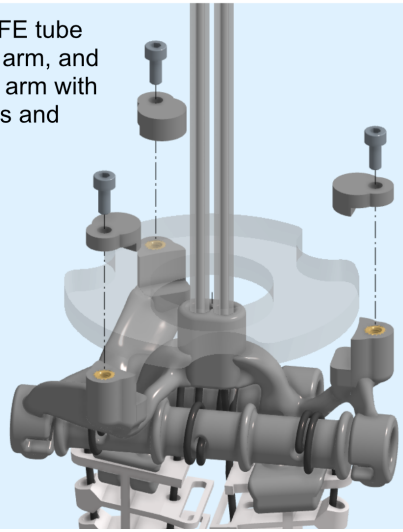
Soft arm end-disc



4.1 Solder the brass inserts into place, and get the tendons through the PTFE

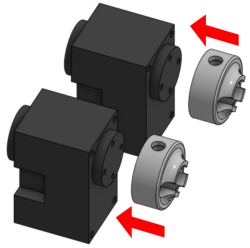


4.2 Fit the PTFE tube inside the arm, and attach the arm with the holders and

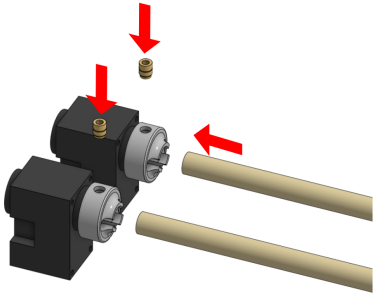


F Assembly: *Actuator*

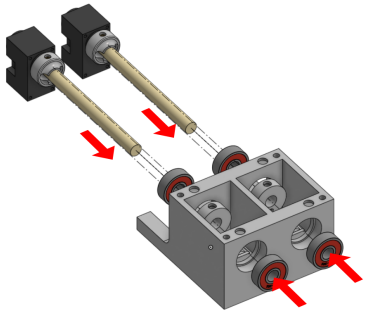
Step 1: Mount horn



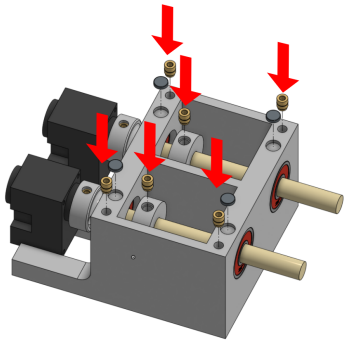
Step 2: Mount axle



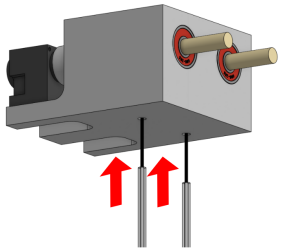
Step 3: Mount axle and bearings to the housing



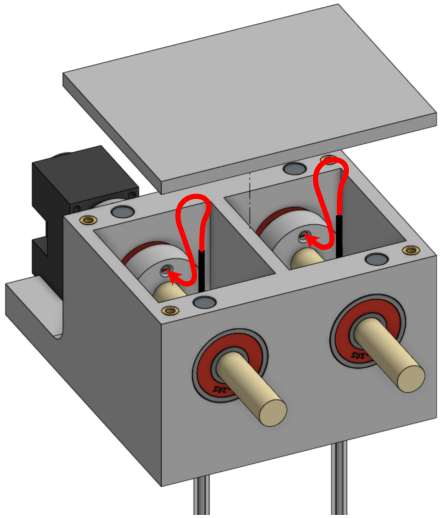
Step 4: Mount Magnets and secure the collar



Step 5: Mount transmission



Step 6: Secure the tendon and close the lid

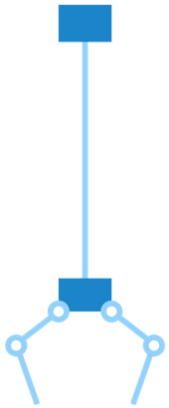


G Bill of Materials

Module	Item	Quantity / length	Price pr unit / meter / kg.	Price gripper
Actuator	Dynamixel XL330-M288-T Servo Motor	2	€ 40.00	€ 80.00
	Bearing 608-2RS	4	€ 0.30	€ 1.20
	Wooden axle 8mm diameter, 1 meter length	2	€ 2.67	€ 5.33
	M2.5 Brass Heat Inserts	6	€ 0.11	€ 0.67
	Magnet 6mm diameter, 1.5mm height	4	€ 0.06	€ 0.24
	Screw M2.5 x 6mm	4	€ 0.01	€ 0.04
Finger mount	Ball Bearing 693-ZZ	4	€ 0.83	€ 3.32
	Screw M2.5 x 6mm	3	€ 0.01	€ 0.03
	M2.5 Brass Heat Insert	3	€ 0.11	€ 0.33
	PTFE tube 4mm OD. 2mm ID.	4	€ 1.60	€ 6.40
	UHMWPE wire 1.5mm diameter, >2 m. length	6	€ 0.63	€ 3.78
	PLA filament spool 1kg.	0.15	€ 20.00	€ 3.00
Finger	TPU 95a filament spool 1kg.	0.014	€ 40.00	€ 0.56
	Silicone rubber A + B mix 1kg.	0.02	€ 40.00	€ 0.80
Total price				€ 105.71

Module	Printables	Quantity		Total price in DKK
Finger mount	Finger mount	1		787.50 kr.
	Large drum half	2		
	Small drum half	2		
	Arm interface top	2		
Finger	Finger 7 g.	1		
	Silicone mold	3		
Actuator	Actuator housing	1		
	Motor coupling	2		
	Tendon coupling	2		

H Decision Matrix: *Deciding the focus of bachelor thesis*



Scale goes from 1 to 5, where 5 is best

Actuator

Factor	Servo Motor	Pneumatic actuator	Weight	Weights for cost focus	Weights for strength focus	Notes
Strength	5	4	4	4	5	You can always buy stronger motors, but the strength for pneumatics is limited by many factors, that all need to be accomedated.
Speed	5	4	2	2	3	-
Ease of control / Sensing	4	5	3	3	0	Servos need more control and sensing, but also has more precision. Pneumatics can limit the need for control
Cost	4	3	0	5	0	The cost for air compressor + valve can be higher than for motors
Miniaturizeability / bulkiness	5	3	3	3	1	The setup for a pneumatic system needs a compressor + valve, whereas for a motor, it can be comparatively the same size as just the valve.
Weighted sum	4.75	4.00	4	4	4	

Transmitter

Factor	Tendons	Pneumatics	Weight	Weights for cost	Weights for strength	Notes
Cost	4	5	0	5	0	With pneumatics you need connecting can be more expensive, due to the cost of manifolds
Energy loss	3	5	4	4	5	Tendons in a "bowden-tube" has more friction than pneumatic tubing
Integration complexity	5	3	3	3	2	The need for air tight in pneumatics sealing can introduce complexity
Weighted sum	3.86	4.14	3	3	3	

Gripper

Factor	Tendon-driven gripper	Flexible Fluid Actuator	Weight	Weights for cost	Weights for strength	Notes
Weight	5	4	5	5	5	Weight is entirely dependent on the design, however FFAs need an air chamber thats entirely sealed which could cause more material and weight
Energy loss	4	3	4	4	5	FFAs consume more and more strain energy the more it's deformed. Tendons generally don't. Therefore tendons can in theory provide more strength
Contact stress on fragile items	2	5	4	4	1	The tendon driven concept needs very clever design to distribute pressure just as well as a FFA
Ease of mechanical design	3	5	1	1	1	↑↑↑↑
Integration complexity	5	3	4	4	2	Sealing issues in FFAs
Safety	4	5	2	2	0	It is possible easier to get pinched by a tendon driven hand.
Cost	5	5	0	5	0	
Weighted sum	4.00	3.95	5	5	5	

Overall

Overall	Tendons	Pneumatics
Weighted score	4.21	4.01
Weighted score with strength sensitivity	4.35	3.92
Weighted score with cost sensitivity	4.24	4.09

Notes

- Whenever we go from one energy form to another we will have loss and increased complexity. Therefore it would be beneficial to maintain one type of energy form
- It might be more challenging (and fun) to use a servo than pneumatics from an electric point of view

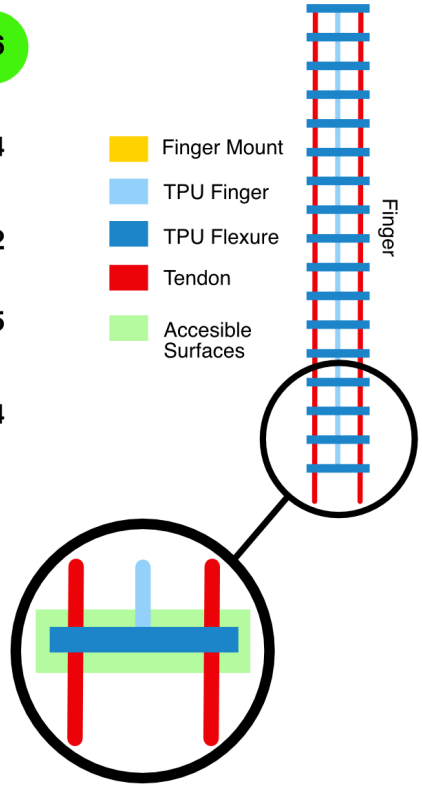
	Sensitivity analysis intermediate calculations	
Actuator with strength weights	5.00	3.89
Transmission with strength weights	3.57	4.43
Gripper with strength weight	4.29	3.64
Actuator with cost weights	4.53	3.71
Transmission with cost weights	3.92	4.50
Gripper with cost weight	4.20	4.16

I Decision Matrix: *Finger mount - finger fastening*

Decision matrix

Type	Weight	Disassembly	Assembly	Structural integrity	Total
Slot	3	5	5	3	16
Screws	3	3	3	5	14
Click-in-place	1	3	5	3	12
Velcro	4	5	5	1	15
Glue	5	1	3	5	14

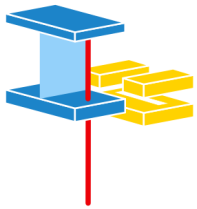
Higher score is better



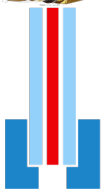
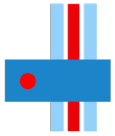
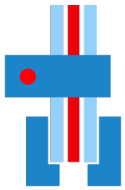
Detailing concept

We chose the slot feature, even though other other options might be more lightweight, as it allowed for easier prototyping, as we could just quickly swap out and try new fingers with this mount.

We made 3 concepts for the finger mount and chose the middle one as we assumed it to be the most lighweight option.



J Decision Matrix: *Finger mount - Transmission mounting*

Type		Actual/estimated Weight	Decision matrix: Weight score	Disassembly	Assembly	Structural integrity	Total
Manifold PC4-M6		4.5 g	**1**	**5**	**5**	**5**	**16**
Friction fit slot		~1 g	**5**	**5**	**5**	**3**	**18**
Clamp with screw		~1 g	**5**	**4**	**4**	**3**	**16**
Clamp with slot		~2 g	**3**	**4**	**4**	**5**	**16**

K Prototyping metal flexure as means for extension

To be able to control the gripper, using only one wire. We thought to use metal, as a reliably torsion spring, instead of the TPU, due to the lack of the viscoelastic effect in metals. This would in theory make the springiness not degrade over time, and make the gripper have a bigger life span. Since we had access to metal wire, we used this as a basic test.

We first tried using silicone as the pulley and surface material, however, it proved far too soft to function effectively, as a finger pulley capable of holding the tendon. We therefore tried using TPU pulleys combined with 6-DOF TPU flexures (designed using FACT), which were used to space out and secure the pulleys position. This was necessary because TPU adhere poorly to metal surfaces and therefore require additional structural support from the flexures to maintain correct positioning.

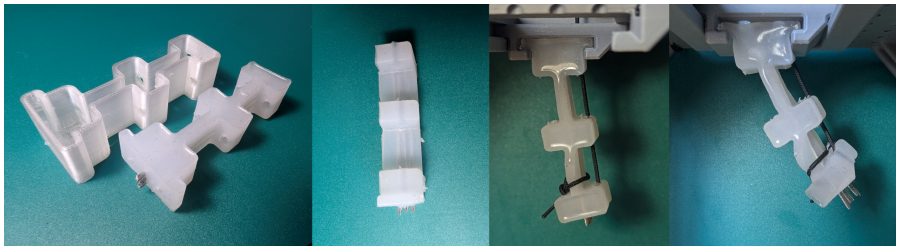


Figure 70: The first silicone joint prototype. Using metal wire for its predictable stiffness behavior. The silicone pulleys fail to translate the applied tendon force into torsion, and the joint does not bend.

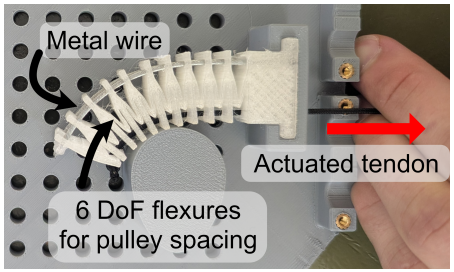


Figure 71: Metal flexure with 6-DOF TPU flexures being actuated

The structural integrity of the setup was excellent. The torsion springs demonstrated good spring-back speed and repeatability, and flexion strength was satisfactory, indicating that the actuation functioned as intended. However, conformability was insufficient. Because the spring tends to minimize its internal stress, it distributes bending stresses along its entire length, making it difficult to achieve local bending greater than the global curvature. As a result, the spring-based extension mechanism did not meet our product requirements, and we subsequently transitioned to a two-tendon system in which both flexion and extension are controlled by tendons.

L Initial grip detection

This graph shows the initial measurements and filtering that we used to detect a grasp while using the servo motors in current based position mode.