

# POLITECNICO DI TORINO

Master Degree in Ingegneria Meccanica



Master Thesis

## Design of a compliant gripper and inverse kinematics control of a soft robotic arm

Supervisors:

Prof. Aurelio SOMÀ

Prof. Josie HUGHES

PhD. Francesco STELLA

Candidate:

Andrea PEIRONE

October 2024



## **Abstract**

Soft robots are celebrated for their ability to store energy in their bodies, and efficiently releasing them during dynamic repetitive motions. I propose to exploit this capability to generate the first soft continuum manipulator able to perform a pick-and-place task in its dynamic regime. To this end, first the thesis will look at developing a fast and adaptive gripper, then it will look at the kinematics of the Helix soft manipulator and the possible human interactions in order to control it. Finally, the two contributions will be combined to analyse how the control is user friendly and to demonstrate the applicability and repeatability of the work. The thesis was performed at the CREATE Lab, École Polytechnique Fédérale de Lausanne, Switzerland.



# Table of Contents

List of Tables	v
List of Figures	vi
Acronyms	xiii
<b>1 Soft Robotics</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Design Principles . . . . .	3
1.3 Key Technologies in Soft Robotics . . . . .	5
1.3.1 Fabrication Techniques . . . . .	5
1.3.2 Actuation Mechanisms . . . . .	6
1.3.3 Control and Sensing . . . . .	7
1.3.4 Energy Efficiency . . . . .	8
1.4 Applications . . . . .	8
1.5 Soft continuum manipulator . . . . .	10
1.5.1 Helix robot . . . . .	13
<b>2 Gripper</b>	<b>19</b>
2.1 Soft and compliant grippers . . . . .	19
2.2 Bistable element . . . . .	21
2.2.1 Design and Geometric parameterisation . . . . .	23
2.2.2 Characterisation . . . . .	24
2.2.3 Analysis of experimental data . . . . .	28
2.2.4 Cup spring model . . . . .	31
2.2.5 Compliant snap model . . . . .	33
2.3 Gripper design with bistable element . . . . .	38
2.3.1 Overview of solutions . . . . .	39
2.3.2 Final solution of bistable gripper . . . . .	40
2.4 Final design . . . . .	43
2.5 Comparison with bistable solution . . . . .	45
2.5.1 Experimental setup . . . . .	46

2.5.2	Validation . . . . .	46
<b>3</b>	<b>Robot control</b>	<b>49</b>
3.1	Kinematics and PCC model . . . . .	49
3.2	Forward Kinematics . . . . .	51
3.2.1	From tendon lengths to configuration . . . . .	53
3.2.2	From configuration to cartesian . . . . .	55
3.3	Inverse kinematics . . . . .	55
3.3.1	From cartesian to configuration . . . . .	56
3.3.2	From configuration to tendon lengths . . . . .	56
3.4	Control . . . . .	58
3.4.1	Boundaries . . . . .	59
3.4.2	Adaptive stiffness matrix . . . . .	63
3.5	Validation . . . . .	64
3.5.1	Square trajectory . . . . .	65
3.5.2	90° bending trajectory . . . . .	70
<b>4</b>	<b>Results and applications</b>	<b>75</b>
4.1	Pick and place of two cubes . . . . .	75
4.1.1	Experimental setup and task . . . . .	76
4.1.2	Data analysis and conclusions . . . . .	77
4.2	Feeding task in order to compare the soft robot with a rigid robot . . . . .	79
4.2.1	Experimental setup and task . . . . .	79
4.2.2	Data analysis and conclusions . . . . .	81
4.3	Applications and capabilities . . . . .	84
<b>A</b>	<b>Python scripts</b>	<b>89</b>
A.1	Main IK code . . . . .	89
A.2	Functions . . . . .	94
<b>B</b>	<b>Questionnaires</b>	<b>107</b>
B.1	Nasa TLX . . . . .	107
B.2	Tust in automation . . . . .	108
	<b>Bibliography</b>	<b>109</b>

# List of Tables

2.1	Geometric parameterisation of bistable elements. . . . .	24
2.2	parameterisation of bistable elements for the snap model. . . . .	35
2.3	Analysis of maximum force point. Comparison between snap model and experimental data. . . . .	38
2.4	Analysis of snapping point. Comparison between snap model and experimental data. . . . .	38
2.5	Comparison between the 2 grippers. The dimensions of width, depth and height are estimated approximately. . . . .	46
3.1	compression limit conditions for each sections. . . . .	60
3.2	Bending limit conditions for the 1° section. . . . .	61
3.3	Bending limit conditions for the 2° section. . . . .	61
3.4	Bending limit conditions for the 3° section. . . . .	61
3.5	Parameterisation for square trajectories. . . . .	66
3.6	Analysis for square trajectories. . . . .	70
3.7	parameterisation of the starting point for 90° bending trajectories. . .	71
3.8	Analysis for 90° bending trajectories. . . . .	73
4.1	Time in [mm.ss] required to do the task for each user. . . . .	77
4.2	Analysis of the survey for the pick and place of 2 cubes task. . . . .	79
4.3	Analysis of the survey for the feeding task. . . . .	82
4.4	Analysis of the survey for the feeding task using Helix robot. . . . .	83
4.5	Analysis of the survey for the feeding task using UR3 robot. . . . .	83

# List of Figures

1.1	Comparison in terms of capabilities between rigid and soft arm. In (a) it is shown the dexterity, in (b) the precision, in (c) the capability of manipulation whereas in (d) the loading. Image taken from [1]. . .	2
1.2	Comparison and state of the art between continuum and articulated soft robots. Image taken from [4]. . . . .	4
1.3	Approaches to actuation of soft robot bodies in resting (left) and actuated (right) states. a) Longitudinal tensile actuators. b) Transverse tensile actuators. c) Pneumatic artificial muscles. d) Fluidic elastic actuator. Image taken from [11]. . . . .	6
1.4	Different types of control systems. In (a) there is an open-loop solution without any sensors or feedback. In (b) it is represented the scheme of a first level closed-loop while in (c) there is the second level closed-loop control diagram. Image taken from [13]. . . . .	7
1.5	Applications of soft robotics. (a) healthcare, (b) rescue, (c) exploration, (d) biomedical application, (e) agriculture, (f) industrial automation.	9
1.6	Under the assumption of piecewise constant curvature, each section of a soft robot can be defined by three parameters: length $L$ , curvature $\theta = \frac{k}{L}$ and angle of orientation of the plane of bending $\phi$ . Image taken from [18]. . . . .	12
1.7	Helix robot. . . . .	13
1.8	Trimmed helicoids structure. In(a) the design of the trimmed-helicoid (TH) structure. $T$ is the thickness of the helicoid, $W$ is the distance from the trimmed edge to the outer radius $R$ , $N_{helix}$ is the helicoid number in one direction, and $\alpha$ is the helical angle of the outer edge of the helicoid. In (b) The tuning stiffness ratio $\lambda_{stiff} = K_{radial}R^2/K_{bend}$ of the TH structures by varying the trimming area. $K_{axial}$ and $K_{bend}$ are the axial and bending stiffness, respectively. Image taken from [20].	14
1.9	Drawing of the trimmed helicoids structure of the Helix robot. . . . .	15



1.10	Drawing of 1 section with the tendons in three different configurations. In (a) the robot is at rest condition, in (b) the three tendons are equally shorten and so the section result compressed whereas in (c) only the tendon 1 is shorten and causes a bending along its direction.	16
1.11	Drawing of 3 sections at rest position with bowden cables sketch for tendons 1,4,7. They are arranged with 30° of rotation between them and it is the same scheme for tendons 2,5,7 and 3,6,9.	17
2.1	Capability comparison between rigid and soft gripper. The hand in the grey diagonal represents the solutions with a mixture of both rigid and soft materials like biological system. Image taken from [23].	20
2.2	General classification of the three gripping technologies for different object type. Image taken from [25].	21
2.3	Typical force-displacement trend for a bistable structure. Image taken from [26].	22
2.4	Comparison between bistable and monostable behaviour for a grip task.	22
2.5	In (a) there is the design of bistable element, in blue are reported the dimensions useful for the parameterisation. In (b) the bistable element is represented in snapped position.	23
2.6	Compression test setup using Zwick Roell z010 machine.	25
2.7	Compression test in direction 1. (a) the starting position of the upper plate, in (b) it reaches the final position, in (c) the upper plate is back to the original position.	26
2.8	Experimental data of model REF during compression in direction 1. The black point represent the snapping position from the 1° configuration to the 2° one while the red point is when the compression test changes the direction and the machine starts to move upwards.	26
2.9	Compression test in direction 2. (a) the starting position of the upper plate, in (b) it is back to the original position.	27
2.10	Experimental data of model REF during compression in direction 2. The black point represent the snapping position from the 1° configuration to the 2° one while the red point is when the compression test changes the direction and the machine starts to move upwards.	27
2.11	Experimental data of model A and model B during compression in direction 1.	28
2.12	Comparison of experimental data for the parameterisation of the thickness $s$ during compression test. In (c) there are the sketch of the 3 models compared.	29
2.13	Comparison of experimental data for the parameterisation of the angle $\alpha$ during compression test. In (c) there are the sketch of the 4 models compared.	30

2.14	In (a) scheme of the cup spring. In (b) the study model. Image taken from [28]. . . . .	31
2.15	In (a) cup spring model. In (b) the equivalent for the bistable element where in black there is the real geometry while in blue the approximation to the cup spring model. . . . .	32
2.16	Cup spring model using Almen-Laszlo expression ( $P_{L1}$ ) and Curti's one ( $P_{L2}$ ) and comparison with experimental data of Model REF. . .	32
2.17	Comparison between the deformation to highlight why the cup spring model is no longer valid after a certain value of deformation. . . . .	33
2.18	Snapping model. (a) The simplest form of the Mises truss, which features bistable 'natural' (highlighted) and 'inverted' (lightly shaded) equilibrium states. (b) This bistability is lost when an additional, linearly elastic, spring of sufficient stiffness is attached vertically to the point mass. (c) Replacing the vertical spring by a viscoelastic element, modelled as a standard linear solid (SLS), maintains the bistable-monostable behaviour. Image taken from [27]. . . . .	34
2.19	Snap model non dimensional using $s$ parameterisation. . . . .	35
2.20	Snap model using $s$ parameterisation. Comparison between model and experimental data. . . . .	36
2.21	Snap model using $\alpha$ parameterisation. Comparison between model and experimental data. . . . .	37
2.22	Overview of gripper designs. The red and orange components are rigid PLA elements that make up mechanisms and transmission organs. The black PLA component is part of the support structure that allows to hold the tendon in order to actuate the bistable element in both directions. In grey there is the bistable element whereas the compliant part is coloured in green. Both of them are printed in TPU. . . . .	40
2.23	Section of bistable gripper. (1) is the compliant gripper itself, (2) is the cap that allows to make the gripper integral to the bistable element.	41
2.24	Actuation of bistable element in order to control the gripper. The sketch represents the close position. In red there is the tendon that control the closure of the gripper while the tendon in blue is the one that allows to open it. (2) the cap already shown in 2.23, (3) bistable element, (4) bistable element support, (5) actuation rod, (6) roll bearings support, (7) roll bearings. . . . .	42
2.25	Model of the gripper without bistable solution. In red the PLA components that make up the transmission organ and in green the flexible gripper. . . . .	44
2.26	Section of the gripper without bistable solution. (1) is the compliant gripper, (2) is the cap that allows to actuate the gripper (3) is the tendon attachment support. In red it is represented the tendon that control the closing behaviour. . . . .	45

2.27	Compliant behaviour of the grippers and comparison between the 2 solutions. In figure (a) and (c) the bistable gripper while in the figure (b) and (d) the final one. . . . .	47
2.28	Workspace of the final gripper. In figure (a) there is a M3 screw to grasp to highlight the capability of the gripper to grasp very thin objects. In figure (b) there is a scotch with diameter of 132 mm which is close to the maximum size of the gripper. . . . .	48
3.1	PCC soft robot sketch. In (a) there are represented 3 sections of a soft robot with the relatives rotational matrices and the relative reference systems of each section. In (b) a single section of CC with the geometrical dimensions to describe the model. In (c) a planar sketch to highlight better the quantities. Image taken from [18]. . . . .	50
3.2	Representation of the tendons of one section $i$ -th. In (a) there is a 3D representation to visualize the section, figure (b) shows the normal point of view of the section $(i - 1)$ -th in order to illustrate how the tendons are arranged. . . . .	51
3.3	Representation of the q-parameterisation, for the $i$ -th segment of a PCC robot. Panel (a) shows the $i$ -th CC segment. In (b) there is the $(i - 1)$ -th section where the two ends of each arc are connected at a distance $d_i$ from the origin of $S_{i-1}$ . (c) shows the $i$ -th cross section and the relevant quantities are underlined in the figure. Image taken from [18]. . . . .	52
3.4	Representation of cartesian space of one section $i$ -th. The point $P_i$ passes on the central axis and the vector $\hat{p}_i$ is normal to the section $i$ -th. In (a) there is the representation of the 3D section with the translation variables highlighted that allows to determine the position of the point $P_i$ . In (b) there is the local referene system with the rotational variables that identify the orientation of the tip of the section. . . . .	53
3.5	Joystick inputs to control the robot. . . . .	59
3.6	Evaluation of maximum compression for each section. The blue dotted line represents the lower limit of the central line l while the red one is the upper limit. In blue and in red are represented the limit configurations of the robot. . . . .	60
3.7	Evaluation of maximum bending for each section. The blue dotted line represents the lower limit of the central line l while the red one is the upper limit. In blue and in red are represented the limit configurations of the robot. . . . .	61
3.8	Workspace in terms of $L_i$ with respect to bending $\Delta$ . . . . .	62
3.9	Comparison between pose to pose control with and without boundary conditions. In transparency the configuration without boundaries with the same goal as the other configuration. . . . .	63

3.10	Grid of points to evaluate the workspace and the the difference using adaptive stiffness matrix. . . . .	64
3.11	Square trajectory represented in frames. . . . .	65
3.12	Layers used for square trajectories and analysis for every plane. Comparison between goal and real trajectory. In (a) there is the illustration of the three different planes considered in order to show the workspace analysed. (b) shows the real configurations of the robot during the test and the square trajectory is highlighted. (c-e) represent the comparison between the goal and real trajectories in the x-y planes studied. . . . .	67
3.13	Square analysis of trajectory 1. In (b) the zoom allows to highlight the parameters taken into account in order to evaluate the performance.	68
3.14	Square analysis of trajectory 1, offset in z direction. . . . .	69
3.15	90° bending trajectory represented in frames. . . . .	70
3.16	Analysis of 90° bending trajectory. In figure (a) there is the front point of view of trajectory 7. The red arrow represents the direction of the end effector at the start of the bending test whereas the blue one is the final orientation. In figure (b) there is the grid of points tested. . . . .	71
3.17	Analysis of 90° bending trajectory 9. In figure (a) there are the parameters involved in order to evaluate the performance. In figure (b) there is the top point of view to visualize the movement along the y-axis of the end effector. . . . .	72
4.1	Experimental setup of the pick and place of 2 cubes task. (a) The 2 crosses on the higher plane represent the starting position of the cubes whereas the cross on the lower plane is the final target where the cubes must be stacked. The distances are in cm. (b) The task with starting and final position presented in transparency. . . . .	76
4.2	Time vs attempts for the cubes task in a plot to evaluate the time and the ease of use of the robot using the joystick as teleoperation system.	77
4.3	Results of the Nasa TLX survey. In (a) there is the trend of the score for each user while in (b) there is the average of the scores of all users for each question, with the standard deviation displayed. . . . .	78
4.4	Experimental setup of feeding task with Helix robot. (a) Setup with target and initial position. (b) shows the user ready to do the test. The task is illustrated thanks to the starting and final position, the last one in transparency. . . . .	80
4.5	Experimental setup of feeding task with UR3 robot. (a) Setup with target and initial position. (b) shows the user ready to do the test. The task is illustrated thanks to the starting and final position, the last one in transparency. . . . .	80

4.6	Results of the Trust in automation survey. The bar represents the average of the scores of all users for each metric. The black sign represents the standard deviation. . . . .	83
4.7	Kitchen scenarios highlighting assistance for disabled people using wheelchair. The robot is mounted directly on the wheelchair, there are some configurations of the robot to show the task. . . . .	85
4.8	Use of a local and small camera installed in the gripper in order to reach a point not visible from the user point of view showing the deformation capabilities of the arm. . . . .	86
4.9	Teleoperated brushing task with a close interaction between the user and the soft arm. The hairbrush is mounted instead of the gripper at the tip of the robot. . . . .	86
B.1	Nasa Task Load Index. . . . .	107
B.2	Trust between people and automation. . . . .	108



# Acronyms

**CC**

constant curvature

**d.o.f.**

degrees of freedom

**ee**

end effector

**FK**

forward kinematics

**IK**

inverse kinematics

**PCC**

piecewise constant curvature

**WS**

workspace

# Chapter 1

## Soft Robotics

Soft robotics represents a significant shift in the way we think about and design robots. By exploiting the properties of soft materials, these robots offer new opportunities for safe, adaptable and efficient automation in a wide range of fields. Despite persistent challenges, ongoing research and development in materials science, actuation mechanisms and control systems are facilitating the wider implementation of soft robotics. This chapter explores the current state of soft robotics, its key technologies, challenges and potential applications, providing a comprehensive overview of this exciting and rapidly evolving field.

### 1.1 Introduction

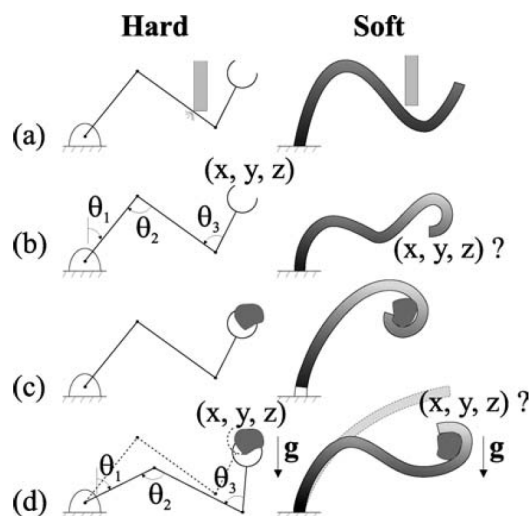
Soft robotics is an emerging field in robotics that focuses on creating robots from highly compliant materials, similar to those found in living organisms. Unlike traditional rigid robots, soft robots are constructed from materials that can deform and adapt to their environment, offering distinct advantages in terms of flexibility, adaptability, and safety when interacting with humans and delicate objects. “Soft” refers to the body of the robot and the materials used for creating soft robot bodies. A good trade-off to define when a robot can be considered "soft" is the Young's Modulus of the materials; even if this parameter is only defined for homogeneous, prismatic bars that are subject to axial loading and small deformations, it is important to take into account because it is a measurement of the rigidity of materials used in robotic systems. For traditional robots the typical used materials, like steels and metals, have moduli on the order of  $10^9$ – $10^{12}$  Pa, while biological organisms are composed of material with moduli on the order of  $10^4$ – $10^9$  Pa as skin, tendons, muscles. A robotic system is defined as "soft" if it is capable of autonomous behavior that is mainly composed of materials with moduli in the range of the ones of soft biological materials.

The motivation behind soft robotics can be attributed to the limitations of traditional



rigid robots, which often struggle with tasks that require delicate manipulation or safe interaction with humans. A traditional robot has typically rigid bodies that are connected by rigid links, they are able to perform specific tasks but providing a single d.o.f. for each configuration, for this reason their ability to adapt and interact with the environment all around is very limited. These robots often encounter difficulties operating in unstructured and highly congested environments. Unlike them, soft robots are constructed from compliant materials, allowing for more flexible and adaptable movements.

The research [1] shows how robots can adapt and react if an external load is applied on them, making a comparison between a traditional and a soft one. For a rigid robot the load makes the joints move but this shift can be compensated using an appropriate control system that can evaluate with high precision the error position of every single joint in order to keep the end effector in a desired pose. For a soft arm, an external load deflects and deforms continuously the robot changing the position of the end effector and all the configuration of the robot itself depending on the design of the arm. This behaviour changes the position of the tip of the arm due to limited sensing and control capabilities but shows other auxiliary characteristics as the larger number of d.o.f. and a large-scale non-linear deformation. All these considerations are visualized in figure 1.1.



**Figure 1.1:** Comparison in terms of capabilities between rigid and soft arm. In (a) it is shown the dexterity, in (b) the precision, in (c) the capability of manipulation whereas in (d) the loading. Image taken from [1].

In medical fields, soft robots can be used for minimally invasive surgeries, prosthesis, and rehabilitation devices that can conform to the human body and provide more natural movement. In manufacturing, soft robots can handle delicate items,

reducing damage and waste. Additionally, in the realm of exploration, such as space and underwater environments, soft robots can navigate complex and unpredictable terrains more effectively than their rigid counterparts. The following sections provide an overview of soft robots, including their design principles, fabrication techniques, and various applications across different domains.

## 1.2 Design Principles

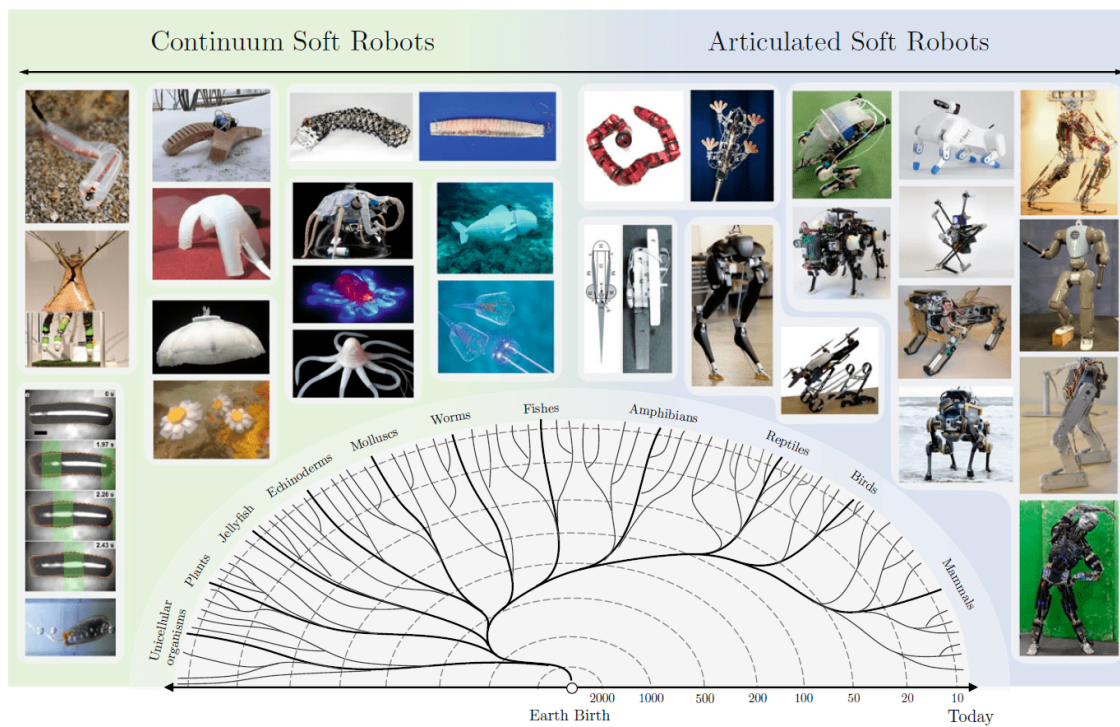
Soft robots are inspired by biological organisms, particularly invertebrates like octopuses and worms, which exhibit remarkable flexibility and dexterity. The design of soft robots emphasizes compliance and deformability, enabling them to interact safely with humans and navigate complex environments. A lot of ideas of soft robots have been developed and they are divided in two main branches: the "Articulated soft robots" and the "Continuum soft robots".

The first ones are built with both soft and rigid parts and they are inspired by vertebrate biological system as mammals, humanoids, reptiles, birds. The idea is to make rigid structure that replicates the skeleton of the robot in order to confer structural stability and use compliant-soft solutions to replicate muscles and tendons typical of vertebrate animals. The soft parts of this type of robot are used mainly for transmission as flexible joints and actuation as cables. This kind of robots are designed to make collaborative tasks with human or to replace them, the mixture of rigid and soft components allows to perform a large variety of tasks as vertebrate animals can do like walking on different terrains, collaborative manipulation. Some examples are shown on the right side of figure 1.2.

On the other hand Continuum soft robots try to replicate invertebrate systems such as snakes and tentacles. The research in this field was born in the 1990s with the first attempts to make continuum structure able to perform with high deformation. In the research made in 1991 [2] a new flexible microactuator driven by an electro-pneumatic system was developed in order to explore the capabilities and possible applications of this new emerging field of robotics. All the kinematics of the continuum prototype was developed to control the 3 d.o.f. of the prototype: yaw, pitch and stretch, which are suitable for miniature robot arms. The main advantages of this solution shown in the article are the easiness to make scaled systems because of the simple structure, the cheapness to fabricate them, the high power density and the smooth movement due to frictionless mechanisms. On the other hand all the kinematics and the control systems beyond this robots are for sure more difficult in respect to the traditional ones. Robinson [3] explored a lot of several forms to realize a continuum robot and they are classified as 'intrinsic', 'extrinsic', or 'hybrid', according to the method and location of mechanical actuation. The main problem is the control because the entire structure undergoes elastic deformation and there are no joints to control or measure displacements in a continuum system. From a technical and design point of view,

the key design principles include:

- **Soft Actuators:** Soft robots utilize pneumatic or hydraulic actuators made from elastomeric materials such as silicone or rubber. These actuators can deform and recover their shape, enabling smooth and continuous motion.
- **Modular Construction:** Soft robots are often composed of modular components that can be assembled and reconfigured to achieve different functionalities. This modular approach allows for rapid prototyping and customization.
- **Distributed Sensing and Control:** Soft robots incorporate distributed sensors and control systems to monitor deformation and adjust behavior in real-time. This enables autonomous operation and adaptive responses to changing environments.



**Figure 1.2:** Comparison and state of the art between continuum and articulated soft robots. Image taken from [4].

## 1.3 Key Technologies in Soft Robotics

The development of soft robotics relies on advancements in several key technologies:

### 1.3.1 Fabrication Techniques

Soft robots are typically made from polymers, silicones, and other flexible materials that allow for significant deformation without damage. Innovations in materials science, such as the development of self-healing materials and shape-memory polymers, have greatly expanded the capabilities of soft robots. Advanced fabrication techniques such as 3D printing, soft lithography, and mold casting play a crucial role in the construction of soft robots. These methods allow for the precise creation of complex, multi-material structures that are essential for the functionality of soft robots. Fabricating soft robots presents unique challenges due to the unconventional materials and complex geometries involved. Several fabrication techniques have been developed to address these challenges:

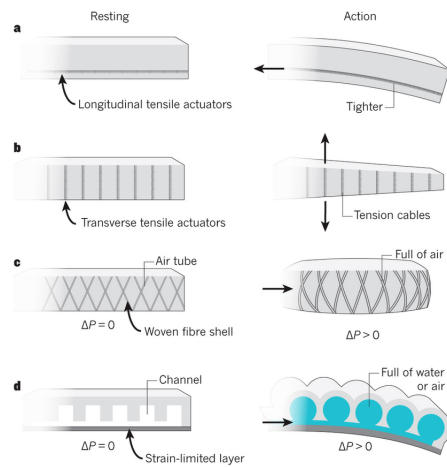
- **3D Printing:** Additive manufacturing techniques such as 3D printing enable the fabrication of intricate soft robot components with high precision and customization. Selective deposition of elastomeric materials allows for the creation of soft actuators and structures. For this fabrication method there are a lot of different techniques suitable for soft robotics [5] such as fused deposition modelling, stereo lithography, shape deposition modelling, multi jet fusion. 3D printable materials used in soft robotics are mainly dielectric elastomers, shape memory polymers, hydrogels, shape memory alloys and fluidic elastomers.
- **Molding and Casting:** Molding and casting techniques are commonly used to produce soft robot components with specific shapes and properties. Silicone molds are created using 3D-printed or CNC-machined master molds, and liquid elastomers are poured into the molds to form the desired structures [6] and also new low-cost injection molding method has been developed [7] to make this technique more usable. With this technique it is also possible to create self-healing materials and shape-memory polymers for soft robotics applications.
- **Soft Lithography:** Soft lithography techniques, such as replica molding and micro-molding, are employed to fabricate soft robot components at microscale resolutions [8]. Photolithographic methods are used to pattern elastomeric materials on substrates, enabling the creation of microfluidic channels and actuators.

All these techniques can be combined in order to create a large range of robot bodies with heterogeneous materials. In addition, the materials used in soft robotics must be durable enough to withstand repeated deformation and use in various environments.

Ensuring the longevity and reliability of soft robots, especially in harsh or demanding conditions, is crucial for their widespread adoption.

### 1.3.2 Actuation Mechanisms

Soft robots require unique actuation mechanisms that differ from traditional motors and actuators used in rigid robots. Common actuation methods include pneumatic and hydraulic systems, shape-memory alloys, tendon cables and electroactive polymers depending on the application [9] [10], each offering different advantages in terms of control, strength, and flexibility.



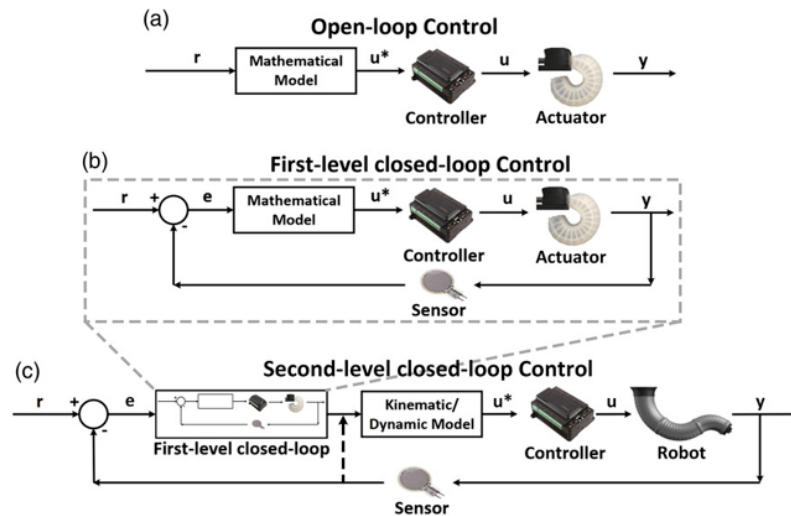
**Figure 1.3:** Approaches to actuation of soft robot bodies in resting (left) and actuated (right) states. a) Longitudinal tensile actuators. b) Transverse tensile actuators. c) Pneumatic artificial muscles. d) Fluidic elastic actuator. Image taken from [11].

For an octopus robotic arm, the tendon solution is the more suitable because it allows to change the shape of the PCC when the tendon is pulled, changing its length. This is the case shown in figure 1.3(a-b). For solutions that try to replicate muscles such as the McKibben artificial muscles [12], the pneumatic approach is the more common one; it consists of an elastomeric tube in a woven fiber shell, and its behavior is shown in 1.3(c). When pressure is applied, the fiber expands radially and it causes a longitudinal contraction of the structure. Fluidic elastomer actuators are built with a strain-limited layer on one side with a number of chambers that can grow up, due to an increase of pressure, changing the shape of the entire structure, making it bend as shown in 1.3(d), and this solution allows to keep this position without additional energy. This type of actuation can be either pneumatic or hydraulic. Regardless of the actuation method, these soft actuators are often configured in a biologically inspired agonist-antagonist arrangement, similar to muscles, to enable

bi-directional control. Additionally, this arrangement allows for adaptable compliance through the co-contraction of muscle pairs.

### 1.3.3 Control and Sensing

Controlling the movement of soft robots is complex due to their highly deformable nature. A rigid robot has a range of determined movements, typically 6 d.o.f. (three rotation and three translation) while a soft robot has infinite d.o.f. because its movements cannot be confined to simple and planar motion. For this reason, controlling this type of robot is very challenging, and developing accurate and responsive control systems that can handle the continuous deformation remains a significant non-trivial task[13]. Compliance structures allow the robots to adapt their shape in the presence of an obstacle of unknown geometry and this kind of interactions generates issues in terms of controlling the soft robots. For this reason new dynamic models, low-level control and planning algorithms have been developed to take into account the presence of uncertainty. On the other hand a compliant structure, used for example as a gripper, allows to do easy pick and place tasks without being so precise in terms of positioning and allows to grasp objects with different sizes and shapes.



**Figure 1.4:** Different types of control systems. In (a) there is an open-loop solution without any sensors or feedback. In (b) it is represented the scheme of a first level closed-loop while in (c) there is the second level closed-loop control diagram. Image taken from [13].

Additionally, integrating sensors into soft materials to provide feedback without compromising flexibility is an ongoing area of research. All the traditional sensors such as encoders and strain gages are not able to perform properly with a soft

solution. New embedded systems[14] need to be developed to provide close-loop control and make soft robots intelligent and autonomous. In order to reach this aim, a lot of sensors are developing and they can be sensitive to different physical quantities such as temperature, pH, chemicals, pressure, light and sound, which significantly depends on the scope of the application. For the deformable structure of the robot itself, also the sensors should be capable to adapt to different shapes without altering the read of the data. This behaviour causes nonlinearities in the system and the result is that there is no a perfect and unique mapping of the values read from the sensors with the real quantities.

### 1.3.4 Energy Efficiency

Soft robots often require continuous power to maintain their shape and perform tasks, leading to concerns about energy efficiency. Developing energy-efficient actuation systems and power sources is essential for the practical deployment of soft robots in real-world applications. For applications that require pneumatic energy sources, a standard fluidic power source (compressors or pumps and cylinders) may not be suitable for soft applications due to its large size and weight, whereas for soft robotic applications the energy sources need to be light, flexible and soft. A lot of new solutions have been explored and compared [15] in order to find a trade-off between flow rates, capacity and maximum pressure using battery-based microcompressors and cylinders of high-pressure fluid. This solution has some issues since the micro-compressors are not energy efficient and the cylinders of little size does not ensure longevity. The research is still working trying to develop portable energy sources or pneumatic batteries.

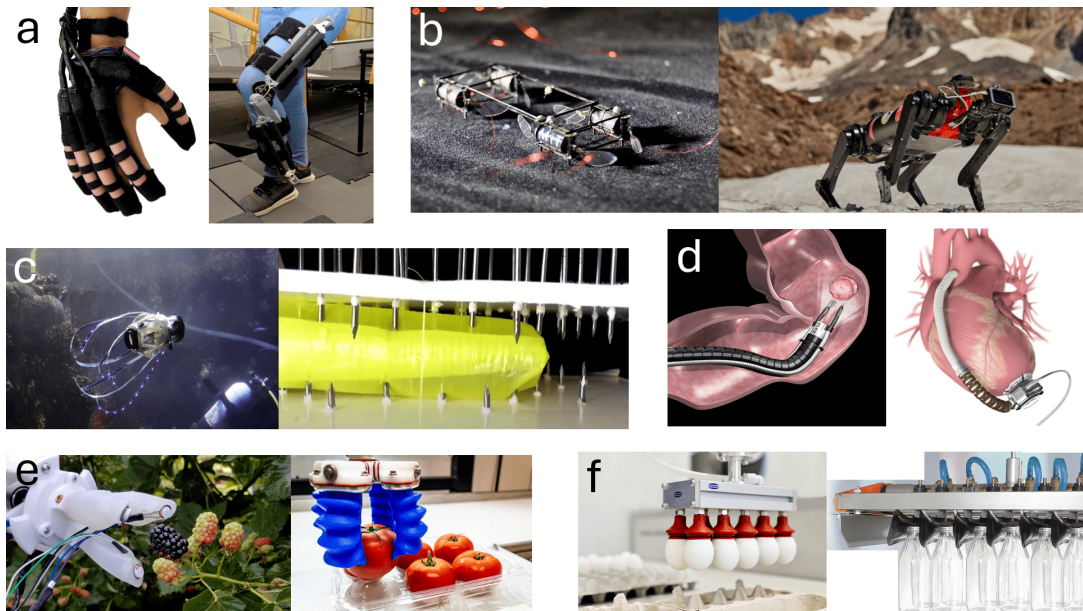
## 1.4 Applications

Soft robots have a wide range of applications in various fields, including healthcare, search and rescue, and human-robot interaction. Some notable applications include:

- **Healthcare:** Soft robots have the potential to revolutionize healthcare through applications such as wearable exoskeletons for rehabilitation or assisting mobility-impaired individuals, assistive devices for individuals with disabilities or age-related mobility issues. This kind of applications can be useful in the everyday-life providing support and assistance, enhancing the quality of life for users.
- **Medical Robotics:** In the biomedical field, soft robots are very suitable thanks to their ability to safely interact with the human body due to their structure. Soft robots as surgical instruments are used in minimally invasive surgeries and rehabilitation therapies due to their gentle interaction with biological tissues such as soft endoscopes for gastrointestinal procedures. Moreover the compliance

of soft robots make them potentially ideal for artificial muscles, ventricular assistance devices that have been already developed in order to replace or help real biological organs. [16].

- **Rescue:** Soft robots can navigate through confined and complex environments, making them suitable for search and rescue missions in disaster-stricken areas. Their flexibility allows them to move through debris and reach locations that are inaccessible to traditional robots.
- **Exploration:** Some solutions are ideal for navigating confined spaces and rough terrain, making them valuable for exploration and surveillance missions. They can traverse uneven surfaces, swim in deep ocean and squeeze through narrow passages, making them well-suited for tasks like pipeline inspection and disaster response.
- **Agriculture:** In agriculture, soft robots can be used for tasks such as harvesting fruits and vegetables, which require gentle handling to avoid damage. Soft robotic grippers can adapt to the varying shapes and sizes of produce, improving efficiency and reducing waste.
- **Industrial automation:** In industrial settings, soft robots can handle delicate and irregularly shaped objects, improving automation in processes that involve fragile components. Their adaptability and gentle touch make them valuable for tasks that require a high degree of precision and care.



**Figure 1.5:** Applications of soft robotics. (a) healthcare, (b) rescue, (c) exploration, (d) biomedical application, (e) agriculture, (f) industrial automation.



Soft robots represent a promising direction in robotics research, offering new possibilities for safe and versatile interaction with the environment. By leveraging compliant materials and innovative design principles, soft robots are poised to revolutionize various industries and domains, from healthcare to exploration. Continued advancements in fabrication techniques and control systems will further expand the capabilities and applications of soft robots in the future.

## 1.5 Soft continuum manipulator

Soft continuum manipulators are one of the specific field of soft robotic in which the research is focused nowadays in order to make robotic arms able to perform collaborative tasks. The first researches and interests on the possibility to develop a continuum soft robot were done at the end of the last century and Robinson was one promoter [3]. The main idea is to build a structure able to perform in the 3d space with a continuum shape inspired by snakes, elephant trunks and tentacles. In this way, the robot can achieve infinite configurations that correspond to infinite d.o.f. and shapes of the system even if the tip of the robot is in the same position. In the last years the research has made a lot of improvements and very several solutions have been developed [10] [9]. This section provides a quick overview of the key technologies discussed in the previous section 1.3 to show the current state of the art of continuum manipulators in terms of technologies :

- **Actuation methods:** The most common method to actuate a continuum manipulator is the fluid-driven mode where the motion of the robot is controlled applying pressure that deforms and alters the shape of the soft material. The typical advantage is the power-weight ratio compared to other solutions while the issues are problems of leakages and potential risk in case of failure due to the pressurized system. Two main solutions has been developed: pneumatic and hydraulic solutions. The first one is preferred and most used for the capability to be pollution free but the liquid solution perform better at high frequency due to the in-compressibility of the material.

Another cheap and popular solution is the cable-driven mode, which involves transmitting the movement by means of a cable that acts as a tendon and can be actuated by a standard motor. The tendon is typically attached to a motor from one side and to an actuated joint from the other one. This type of actuation is integrated within the soft body, and, in comparison with the previous solution, is more autonomous due to the absence of requisite pressure sources, such as compressors or pumps. The main issues with tendon-driven solution are the non-linear frictions, non sufficient transmitted force and stretching phenomena that can alter the total length of the cable. On the other hand it is a safer solution because in the worst case of failure of broken tendon, the possible user in the same working area of the robot is not seriously injured.

In order to solve some problems of the previous and standard solutions, new actuation method has been developed such as Stimulus-Responsive Actuation [9]. Physical stimuli, such as electrical, magnetic, or light, are applied to the structure in order to control the movements and the shape of the robot, for this reason are required "smart materials" embedded in the soft bodies that react properly to external stimuli. One much investigated category involves the shape memory alloy (SMA), with the properties of the soft materials is possible to deform the shape under external force and recover the original memorized shape if upon heating due to the material characteristics such as high elongation and temperature sensitivity. If the external stimulation is electrical or magnetic the actuation mode is called Electroactive polymer (EAP) and it can be divided into ionic types and electronic type. The first ones are activated by anelectrically induced diffusion of ions and/or molecules within the bulk of the material, whereas the electronics are activated by Coulomb forces.

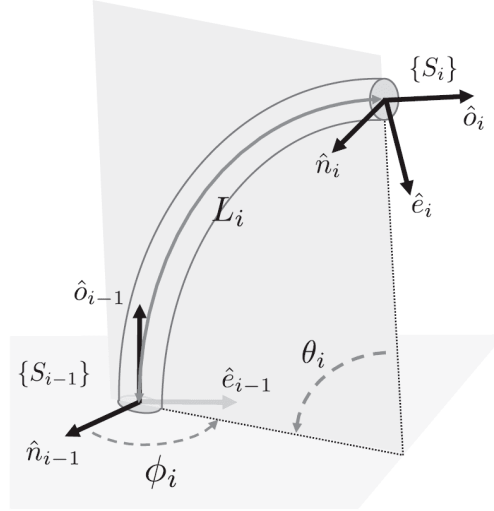
- **Model control:** Motion control defines the relationship between the variable control and the shape of the continuum robot. Model-based control is used in cases where a model control has been developed based on kinematics and dynamics laws. The aim is to obtain a transfer function that allows a relationship to be found between the control variables, which are the inputs, and the configuration of the robot, which is the output.

On the other hand, model-free control refers to motion control without depending on physical model but the control can be based on vision and intelligent algorithm. For this reason is easier and quicker to implement but with a higher degree of uncertainty and find the most of applications in robot prototypes. The control of the soft manipulator depends on the robot itself in terms of design ad fabrication mode.

- **Kinematic approaches:** To model a soft continuum arm, it is essential to take into account the mechanical deformations along the soft body, which make modelling non trivial. The first and easiest approach is the constant curvature one, it consist on approximating the body of the robot into an arc of constant curvature with mathematical and geometrical relationships. In this way, the implementation is relatively simple and allows the shape to be quickly evaluated. This approach was widely used in the early years of research, but is no longer suitable for the complex geometries and designs of the new continuum manipulators.

In order to make the model closer to real robot arms and maintain a relative easiness in calculations, the piece-wise constant curvature (PCC) model has been developed for the first time in 2006 by Walker [17]. It consist of approximate sections of the robot to arcs with constant curvature as the previous model but the entire body can be arranged into more complex shapes combining different sections. In figure 1.6 a PCC sketch for one section is showed, the analysis

of these quantities and the corresponding kinematics will be developed in the chapter 3.



**Figure 1.6:** Under the assumption of piecewise constant curvature, each section of a soft robot can be defined by three parameters: length  $L$ , curvature  $\theta = \frac{k}{L}$  and angle of orientation of the plane of bending  $\phi$ . Image taken from [18].

Also non constant curvature methods have been developed in order to abandon the assumption on constant curvature and make mathematical models that can evaluate locally the shape of robotic arm. The first example implies the classic dynamic model mass-spring-damper. The aim of this approach is to evaluate the kinematic characteristics and to easily perform frequency analysis, the tricky part is to model a soft robot with equivalent parameters such as mass and spring that behaves in the model as the real robot. New mathematical models have been applied in continuum arm in order to overcome these problems such as the cosserat geometrically exact model. Grazioso et. al. [19] made a discrete model using a helicoidal shape function for the spatial discretization and a geometric scheme for the time integration of the robot shape configuration.

- **Shape detection technologies:** In contrast to a conventional robotic arm, the evaluation of the configuration of a soft continuum in real time is a challenging task. This represents a significant obstacle to the implementation of a closed-loop control system. As discussed in chapter 1.3.3 new technologies have been developed in order to build sensors able to give a direct feedback and they can be divided in two main branches: embedded systems or external devices. The embedded shape detection method for the soft arm is still in the initial stages of development. Being independent from external disturbances, being

properly distributed along the soft body and give an accurate measurement of the variables taken into account are the main technical challenges for this solution. Also from a fabrication point of view is hard to find sensors able to perform in the deformation domain typical of soft arms. Inertial measurement unit (IMU) are small devices that basically can evaluate the orientations of the sensor along the three main cartesian axes and can be useful in some continuum manipulators.

3D vision measurement is another solution to evaluate the real time shape of the robotic arm and the position of the end effector. The main detection system is the optical tracking system that consists on applying markers in specific points along the continuum arm that are recognisable by some cameras mounted in the workspace of the robot. After a calibration is possible to record the motion of the markers in the 3D cartesian coordinates.

### 1.5.1 Helix robot

The work of this thesis has been done with the soft continuum manipulator Helix developed at the EPFL CreateLab [20] in figure 1.7. It is a continuum manipulator with PCC solution with 3 sections held by the support structure.

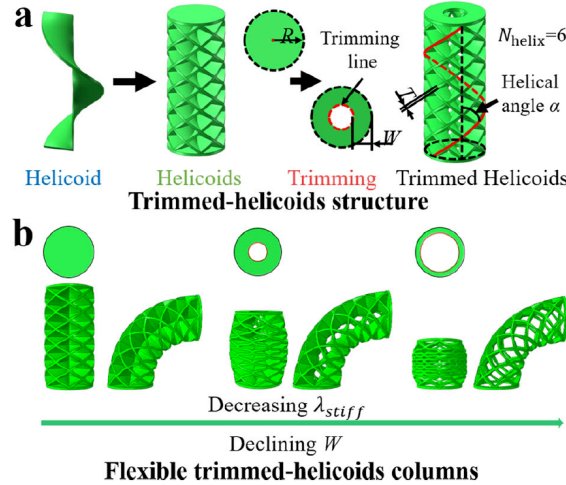


Figure 1.7: Helix robot.

## Structure

The robot itself is made in TPU with a innovative 3D printing technique called multi jet fusion [21]. The sections of the arm is circular with architected structure [20] developed and designed in order to find the right trade-off between controllability, sensitivity to errors in control, and compliance. The structure is scalable and allows to obtain different stiffnesses according to the material, the size and the ratio between width and radius.

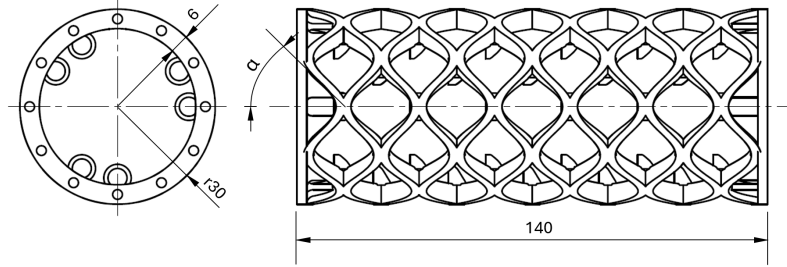
The structure allows to make the bending and the compression, which are the main possible deformations of a continuum manipulator, independent of each other. To reach this goal, a helicoid structure is radially trimmed from the central axis. In this way the outer part of the structure provides bending stiffness to the body while the inner one prevents compression. Tuning the geometric helical angle, which is the only parameter that defines the shape, and the radius of the trimming line, which defines the width, is possible to change the bending stiffness and the axial one depending on the values required.



**Figure 1.8:** Trimmed helicoids structure. In(a) the design of the trimmed-helicoid (TH) structure.  $T$  is the thickness of the helicoid,  $W$  is the distance from the trimmed edge to the outer radius  $R$ ,  $N_{helix}$  is the helicoid number in one direction, and  $\alpha$  is the helical angle of the outer edge of the helicoid. In (b) The tuning stiffness ratio  $\lambda_{stiff} = K_{radial}R^2/K_{bend}$  of the TH structures by varying the trimming area.  $K_{axial}$  and  $K_{bend}$  are the axial and bending stiffness, respectively. Image taken from [20].

The robot used for this work has these features and characteristics: The ratio between width and radius is 0.2, respectively the width is 6 mm and the radius is 30 mm as shown in figure 1.9. The helical angle  $\alpha=52.6^\circ$ , the thickness of the helicoid  $T=3\text{mm}$ . The holes inside the structure are the supports for the bowden cables that allows to control each section independently. At rest the length of the first section is

14 cm whereas the second and the third ones are 28 cm each because they are the union of 2 printed parts of 14 cm. Each section is connected by M2.5 screws and the total length of the arm at rest is 70 cm. The decision to make the first section smaller depends on the weight each has to carry, the first section has to be smaller because of the weight of the remaining parts of the robot.



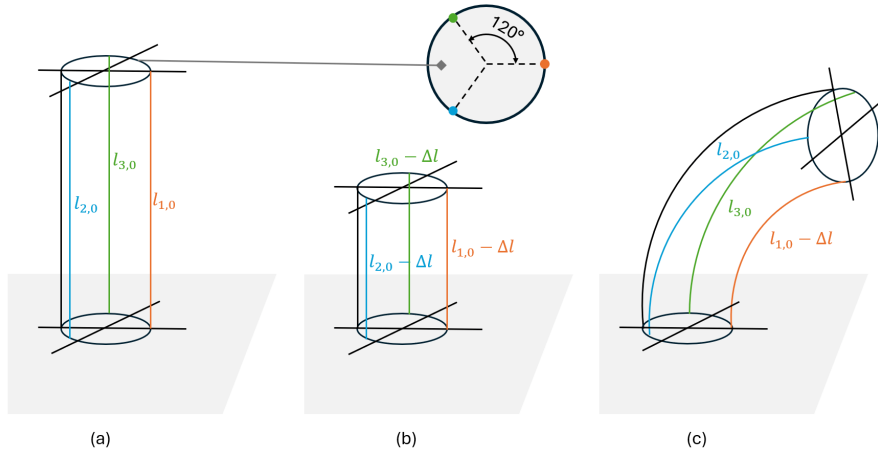
**Figure 1.9:** Drawing of the trimmed helicoids structure of the Helix robot.

### Actuation

The robot is actuated thanks to tendons driven by rotating motors (Dynamixel, XM430-W210-R). Each motor controls one tendon and for each section there are three tendons arranged at  $120^\circ$  so in the overall with 9 motors is possible to move all the arm. The motors are housed at the top of the arm and they are arranged in two layers, three in the lower one that are attached directly to the tendon thanks to a 20mm radius pulley and control the first section while the other six are mounted on the upper layer and are attached to the tendons thanks to bowden cables in order to control the second and the third section. The tendons are arranged symmetrically along the circular profile with an angle of  $120^\circ$  between them 1.10. When a motor is actuated, the tendon shortens and causes the compression of the TPU structure along his direction.

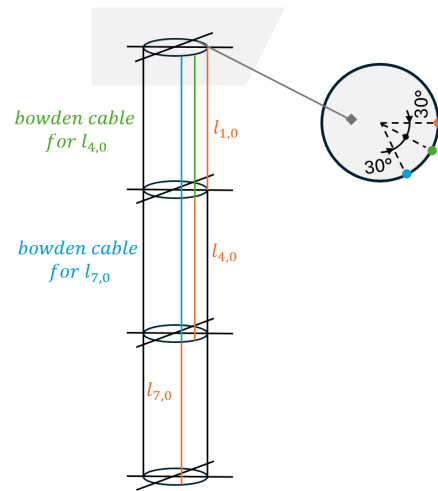
The illustration 1.10 demonstrates the manner in which tendon actuation, whether uniform or selective, can regulate the motion of the robot by generating either compression (uniform shortening) or bending (unequal shortening). This principle is frequently employed in soft robotics and continuum robots in order to achieve flexible and adaptable movement. The robot in resting configuration is shown in (a) the three tendons  $l_{1,0}$ ,  $l_{2,0}$ ,  $l_{3,0}$  are of equal length, thereby maintaining equilibrium in the structure. The tendons are neutral and not shortened with the robot in a upright position. In (b), all three tendons are equally shortened by an amount  $\Delta l$ , causing the structure to undergo uniform compression. In (c) only tendon 1 is shortened by  $\Delta l$ , while tendons 2 and 3 remain at their original lengths. This results in the section bending in the direction of the shortened tendon. The unequal tendon lengths induce bending along the direction of the shorter tendon (tendon 1), as the tension imbalance pulls the structure into a curved shape.. Combining the three

tendon lengths is possible to control the movements of the structure in every possible direction and achieve different shapes of the continuum robot.



**Figure 1.10:** Drawing of 1 section with the tendons in three different configurations. In (a) the robot is at rest condition, in (b) the three tendons are equally shorten and so the section result compressed whereas in (c) only the tendon 1 is shorten and causes a bending along its direction.

This robot is an independent manipulator, which means that the movement of each section is separated from the shape of the others; to reach this goal it is mandatory to make sure that the compression of a single tendon acts only on one selected section. In particular the tendons 1,2,3 act on the first segment whereas the number 4,5,6 control the second one and the 7,8,9 move the last section. To illustrate, if tendon 4 is shortened, only the second section bends, while the first and third remain unchanged. This phenomenon occurs in all cases. To achieve this, a solution utilising Bowden cables is employed, enabling the transmission of tendon length along the structure of the robot, bypassing the previous sections. The solution adopted for tendons 1,4,7 is illustrated in figure 1.11 with the robot in resting configuration. The tendons are arranged in groups, with a  $30^\circ$  rotation between them as indicated by the circular cross-sectional view. The figure also sketches the bowden cables (green and blue lines respectively for bowden cable that house  $l_{4,0}$  and  $l_{7,0}$ ) that house the tendons, allowing for controlled actuation of the robot's sections. Bowden cables provide flexible routing for the tendons while transmitting force. These tubes are made of PTFE (4 x 2 mm), they are fixed to the upper base plate of the motors by means of supports, they pass through a  $180^\circ$  bend and are then housed in the holes visible in the front view in the figure 1.9, which reach the end of the respective section where they are attached with a M2.5 screws.



**Figure 1.11:** Drawing of 3 sections at rest position with bowden cables sketch for tendons 1,4,7. They are arranged with  $30^\circ$  of rotation between them and it is the same scheme for tendons 2,5,7 and 3,6,9.





# Chapter 2

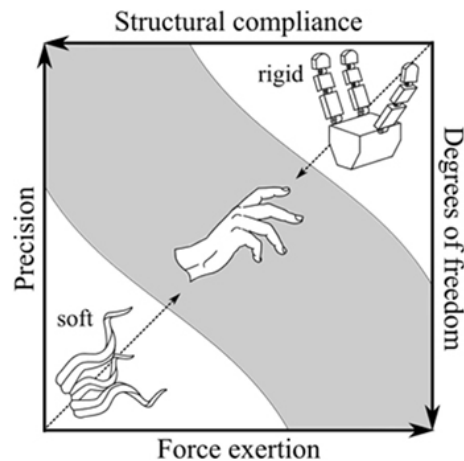
## Gripper

The act of grasping objects is one of the most pervasive tasks in everyday life. In the field of industrial automation, the technologies discussed in the previous chapter, which pertain to the domain of soft robotics, are employed in the design of soft grippers on a multitude of occasions. The capacity for deformation under minimal forces and the ability to assume diverse shapes in response to external stimuli render soft robotics an optimal choice for the engineering of compliant grippers that can be deployed in a multitude of tasks. This chapter provides an overview of the current state of the art in compliant grippers and then presents the two main solutions developed in this thesis for the design of a gripper for the Helix robot.

### 2.1 Soft and compliant grippers

Since soft robotic grippers provide excellent adaptations to different shapes and for a wider range of objects if compared with rigid grippers, a lot of solutions have been developed in the last years of research [22]. Rigid grippers are designed with rigid bodies and can work properly with rigid objects to be grasped through the principle of mechanics, while a soft solution could be more universal for grasping tasks because the large deformation of the soft body increases dexterity and adaptation to objects. The ability of a soft robot to interact safely with the external environment makes these new technologies more suitable for systems that need to grasp objects.

Four parameters are taken into account to highlight the capabilities between a soft gripper and a rigid one: precision, structural compliance, d.o.f. and force exertion shown in figure 2.1. The best solution for a new gripper design between these different possibilities, depends on the specific application and features required for the gripper. In most cases a hybrid design is the right trade-off in order to obtain sufficient closing force using a compliant structure.

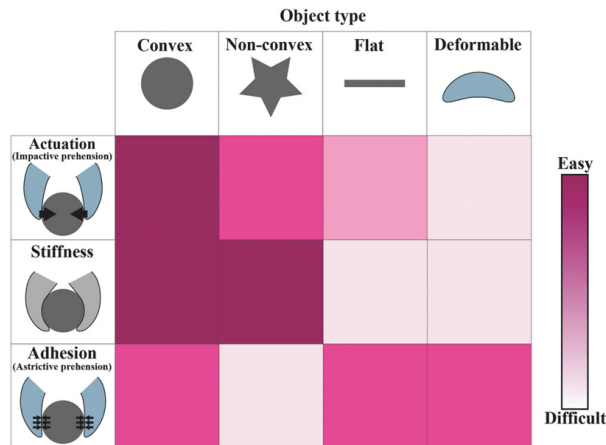


**Figure 2.1:** Capability comparison between rigid and soft gripper. The hand in the grey diagonal represents the solutions with a mixture of both rigid and soft materials like biological system. Image taken from [23].

Soft robotic gripping can be divided in three main technologies for different object properties that are not completely exclusive but they can be combined together to improve the performances [24]:

- **Actuation:** This type of gripping allows objects to be gripped thanks to external actuators that deform the shape of the gripper according to the object to be grasped. The working principle consist on close the fingers or elements around the objects, for this reason it is very suitable for convex and non-convex objects but it can not perform with deformable and flat ones as shown in figure 2.2. This one is an easy and quick solution to develop but, on the other hand, it is not sensitive to the surface, the gripping force required is high and can be not easily controlled. There are several technologies to achieve this behaviour such as motors, fluidic elastomer actuators, electroactive polymers or shape memory alloys as already presented in sections 1.3.2 and 1.5.
- **Stiffness:** This type of gripping control is based on the variable stiffness of the body. The gripper starts in the soft configuration as it approaches the object and then switches to the stiffer configuration as it grips the object. Stiffness can generally be tuned for each application. An actuator is needed in order to close the fingers but since they are soft, to wrap the object a minimum force is required from the actuator if compared with the previous solution. This kind of gripper can perform quick pick and place tasks. An in-depth study of materials science is underway and technologies such as granular jamming, low melting point alloys, electrorheological fluids and shape memory materials are being applied. For the behaviour of the gripper, this technology is recommended for convex and non convex shapes but not for flat or deformable objects.

- **Adhesion:** The working principle of this grasping technology is due to the surface force that the gripper exert on the object that is no as high as the one needed for actuation method so it is recommended for very fragile objects to manipulate. It is possible to hold an object thanks to the shear stress that is proportional with an high ratio to the normal pressure generated. Many types of adhesion control have been developed, such as electroadhesion, which works thanks to electrostatic attraction, dry adhesion, which works with van der Waals forces between the object to be gripped and the soft gripper, and solutions that require a vacuum adhesion to hold an object by means of cups. This solution is ideal for flat or deformable objects with smooth and clean surface, but it is not suitable for non-convex objects because it is very difficult to find a surface that fits with this type of interactions.



**Figure 2.2:** General classification of the three gripping technologies for different object type. Image taken from [25].

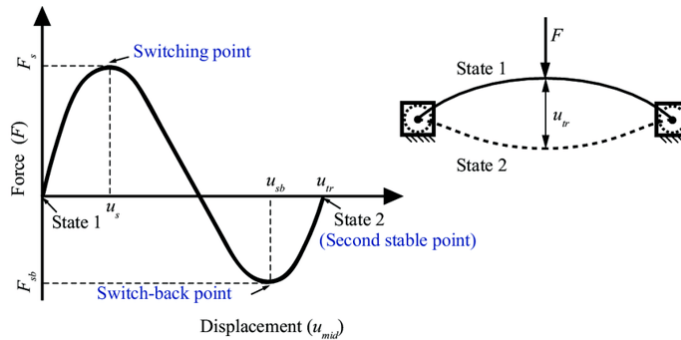
Another classification can be done in respect of the d.o.f. of the gripper. The 1-d.o.f. solution is the most common one and allows to grip the object thanks to a rotation, translation or a combined motion. Multi-d.o.f. grippers have been developed in cases that try to replicate humanoid hands such as multi-fingered dexterous hand. Typically the number of d.o.f. corresponds to the number of actuators but, to simplify the control for multi-d.o.f. grippers, there are also some solutions in which the number of actuators is less than the d.o.f. of the system [24].

## 2.2 Bistable element

The work of this thesis is focused on the development of an innovative soft gripper made for the Helix robot presented in 1.5.1 with a bistable behaviour that allows to

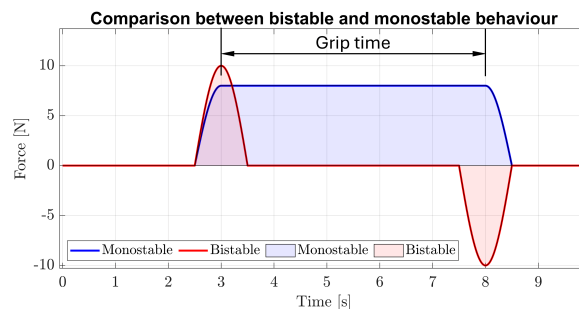
perform fast pick and place tasks and is an energy saving solution for the actuators. This section is about the research of a bistable element, its parameterisation and analysis in order to find a trade-off suitable for the gripper.

The typical trend of the force in function of the displacement is showed in figure 2.3 to highlight the two different stable configurations and how the structure snap between them in terms of applied force.



**Figure 2.3:** Typical force-displacement trend for a bistable structure. Image taken from [26].

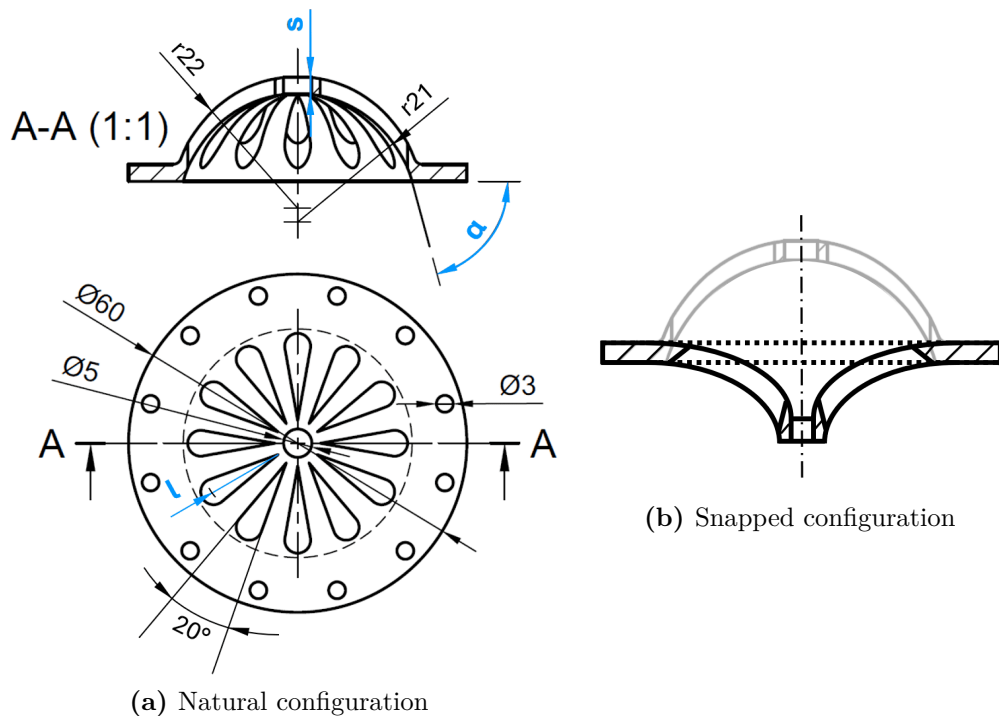
The bistable behaviour of the gripper allows to save energy when an object is grasped. In figure 2.4 is showed the comparison of performance between bistable and monostable solutions applied for a grasp task of 5 seconds in terms of force required by the actuators. The difference is that with the bistable technology, the actuator is in stand-by mode during the grip time while for the monostable one it is always turned on during the 5 seconds. The underlying area in the force-time graph is the work to be done by the actuators and it is evident that the area, and therefore the amount of energy required, is much smaller for the bistable solution. The ratio between the areas of the two different methods increases as the grip time increases. The graph presented in 2.4 is theoretical just to highlight the differences between the two solutions explored in this thesis.



**Figure 2.4:** Comparison between bistable and monostable behaviour for a grip task.

### 2.2.1 Design and Geometric parameterisation

To design the bistable gripper, the first step is to create an element that can snap between two equilibrium positions. Many solutions have been developed in literature in order to build and model snap elements [27], after several attempts the final design has a dome shape with a behaviour inspired by mechanical cup spring [28]. The aim of this design is to obtain an element that is asymmetric in the behaviour, which means that the force to make it snap in one direction is higher than the one required in the other direction. This behaviour is necessary for the production of a final gripper that requires a low force to close and an higher one to open it, in this way the closing force should be higher and at the same time the force to open it is low. All the models have been 3D printed with FDM technique using Prusa mk4 printer and many attempts have been done with different TPU filaments to find the desired characteristic in terms of stiffness and surface quality. "Flex medium" filament by Extrudr has been used with this main features from datasheet: Young Modulus  $E = 40$  MPa, ultimate elongation 475%. The final prototypes have been realised with NinjaFlex Filament 85A by Ninjatek with these features: Young Modulus  $E=4$  MPa, ultimate elongation 660%. The print parameters used are  $T_{nozzle}=235^\circ$ ,  $T_{bed}=30^\circ$ , layer height=0.15 mm, print speed=60 mm/s.



**Figure 2.5:** In (a) there is the design of bistable element, in blue are reported the dimensions useful for the parameterisation. In (b) the bistable element is represented in snapped position.

This model has three main variables to characterise the final shape: the angle  $\alpha$  at the bottom line, the thickness  $s$  at the central line and the length  $l$  of drop-shaped holes. The dimension of the external circle is defined by the diameter of the section of the robot and so the 12 radial holes with the diameter of 3 mm for the screws. The internal and external radii of 21 and 22 mm are defined in order to ensure a vertical displacement coherent with the application. The dimension  $l$  of the 12 drop-shaped holes is evaluated in order to make the hole tangent to the junction and allows the structure to snap as shown in figure 2.5(b). In the final configuration, the gap between these holes is filled because the two sides are in contact. The central hole has to support the actuation part when this solution is applied to the gripper. The geometric parameterisation has been done with 9 different set of values changing the thickness  $s$  and the angle  $\alpha$ . This set of models has been done to test different shapes that change stiffness and total displacement between the 2 stable positions in order to find a trade-off. Starting from the REF model, A-B-C-D are obtained changing the  $s$  parameter with 1 mm step maintaining the angle to a constant value of  $75^\circ$ , then E-F-G-H models are build changing  $\alpha$  from  $60^\circ$  to  $90^\circ$  with  $10^\circ$  steps and maintaining constant the thickness  $s = 3$  mm. The value of parameter  $l$  is evaluated in order to keep the drop-shaped hole tangent to the junction between the flat surface and the dome.

MODEL	$s$ [mm]	$\alpha$ [ $^\circ$ ]	$l$ [mm]
REF	3	75	14
A	1	75	13.5
B	2	75	13.6
C	4	75	14.2
D	5	75	14.5
E	3	60	12
F	3	70	13.5
G	3	80	14.5
H	3	90	15

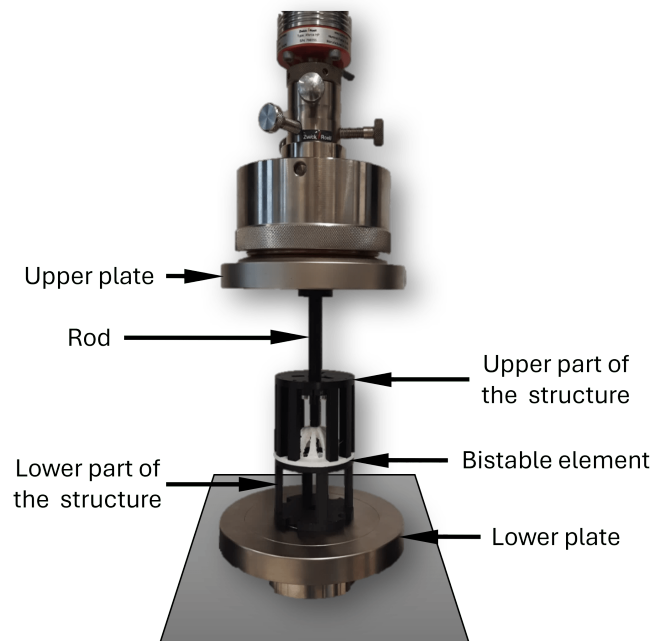
**Table 2.1:** Geometric parameterisation of bistable elements.

### 2.2.2 Characterisation

The objective of the asymmetric design is to construct a passive gripper, which is capable of closing itself without the need for actuation. This is achieved through low-force contact between the gripper and the object, which is initiated upon approach. Prior to constructing the entire gripper, it is essential to characterise the complete set of bistable elements in both directions and ascertain whether the asymmetric behaviour can be obtained from a force-displacement graph. The models have been evaluated for their sensitivity to alterations in the geometric parameters.

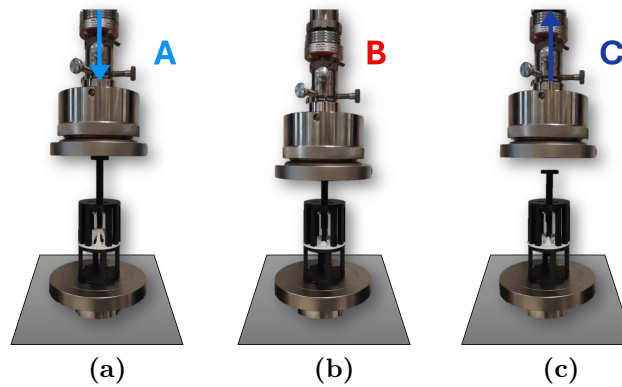
### Setup for compression test

The compression test has been done with Zwick Roell Z010 machine with a maximum test load of 10 kN and a sensitivity of 1 mN. The set up is showed in figure 2.6. The objective of the test is to evaluate the starting and finishing positions of the upper plane while the lower plane is fixed to the machine structure. Following the evaluation of the total displacement, the upper plate commences its downward movement until it reaches the minimum force threshold, which has been set at 0.1 N. After that, it continues its movement at a slower pace, maintaining a constant speed of 0.5 mm/s, in order to test the system in conditions approaching static equilibrium until it reaches the final target. Finally, the upper plate begins its upward movement to the initial position, continuing to record the values of force and displacement. The bistable element is affixed to two PLA structures with six M2 screws and a PLA rod situated at the central axis. This configuration enables the transmission of movement from the upper plate to the bistable element. To conduct the test in the opposite direction, the upper and lower components of the structure are interchanged. In this configuration, the rod traverses the lower portion of the structure, thereby exhibiting the same behavior as observed previously.



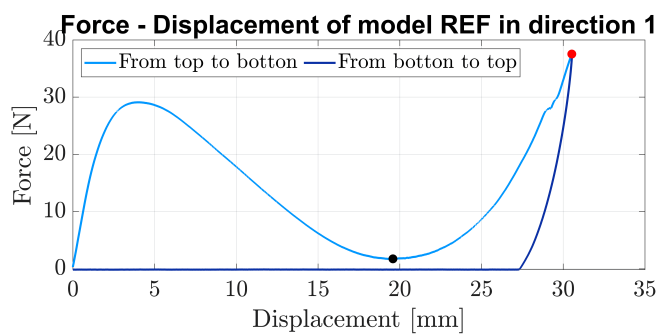
**Figure 2.6:** Compression test setup using Zwick Roell z010 machine.



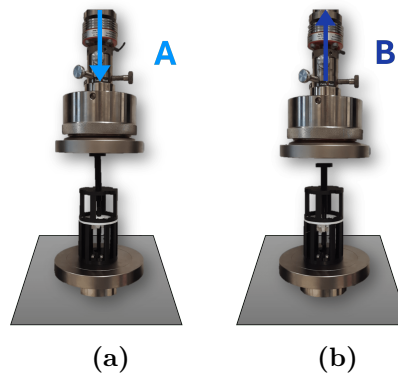


**Figure 2.7:** Compression test in direction 1. (a) the starting position of the upper plate, in (b) it reaches the final position, in (c) the upper plate is back to the original position.

In figure 2.7 there are the three main states of the test in the direction 1 which starts with the bistable element in the natural configuration 2.5(a) and ends in the snapped position 2.5(b). In figure 2.8 is showed the corresponding force-displacement graph obtained for the REF model, the color of the curves are matched with the letters in the previous figure. The bistable behaviour is clear analysing the blue line, which corresponds to the upwards movement of the plate, is zero because after reaching the snapped state there is no more contact between the rod and the upper plate. The increase in force is a consequence of the fact that, in the context of the test, the displacement range is defined (in this case, 30 mm) and the force measured at the point of snapping of the bistable element rises in response to the fact that the force applied is stored within the structure, but is no longer capable of moving the bistable element.

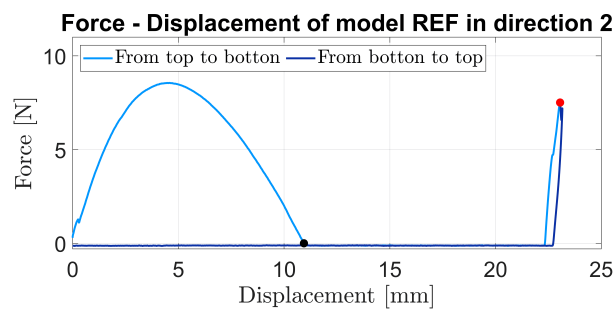


**Figure 2.8:** Experimental data of model REF during compression in direction 1. The black point represent the snapping position from the 1° configuration to the 2° one while the red point is when the compression test changes the direction and the machine starts to move upwards.



**Figure 2.9:** Compression test in direction 2. (a) the starting position of the upper plate, in (b) it is back to the original position.

In figure 2.10 the graph corresponds to the test in direction 2 for model REF. Comparing this characteristic with the one obtained in 2.8, it is clear the asymmetric behaviour because the maximum force decreases from 28 N to 8 N. Also the shape of the curve is different because in direction 2 the displacement of the snapping point is lower and after that, the force is zero because the bistable element has already switched back to the dome shape. This trend is coherent with the passive behaviour because in this direction the amount of force required is significantly lower. The grow of the force in the final part of both the graphs is due to having reached the end of the test range but is not significant in terms of characterization of the models. From the presented graph, it can be discerned that the 30 mm range utilized for the alternative direction is excessively large, as evidenced by the observation that the maximum vertical displacement of the bistable element is approximately 22.5 mm. In order to ascertain the snapping point, it is essential to ensure that the test is concluded even if the total set displacement exceeds the actual value.

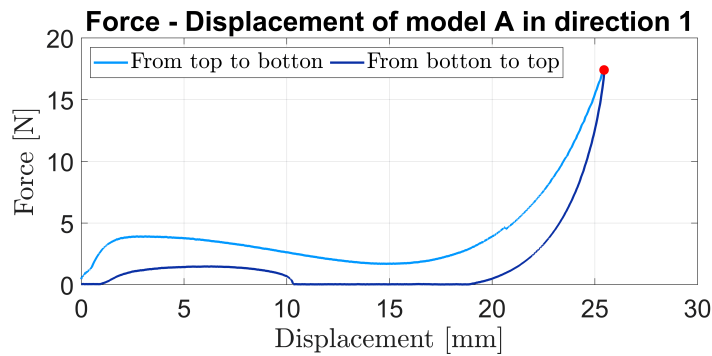


**Figure 2.10:** Experimental data of model REF during compression in direction 2. The black point represent the snapping position from the 1° configuration to the 2° one while the red point is when the compression test changes the direction and the machine starts to move upwards.

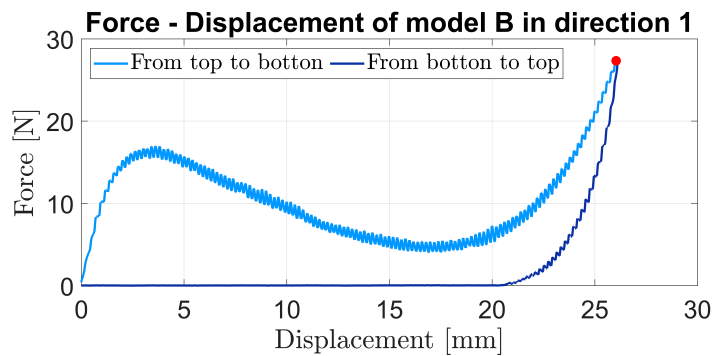
### 2.2.3 Analysis of experimental data

The test described in the previous section has been carried out on each model and, after collecting the data, the model A and the model B (tab 2.1) are no more taken into account for the comparison for two different reasons.

The model A is not bistable because the green line in figure 2.11(a) is not zero from all the upward movement of the upper plate. During the test, the element could not stay in the snapped position and the reason may be that the thickness  $s = 1$  mm does not give the model sufficient stiffness. Instead the trend of model B is not so smooth and there are a lot of spikes during the compression test, maybe this behaviour is due to the non-homogeneous 3D printed material of this model. Many bistable elements have to be printed several times because the 3D printing process doesn't always guarantee good surface quality. Frictions between the rod and the structure could be the problem but this hypothesis is rejected because the structure used in all the tests is the same and the spikes are not present in the other cases.



(a) Model A



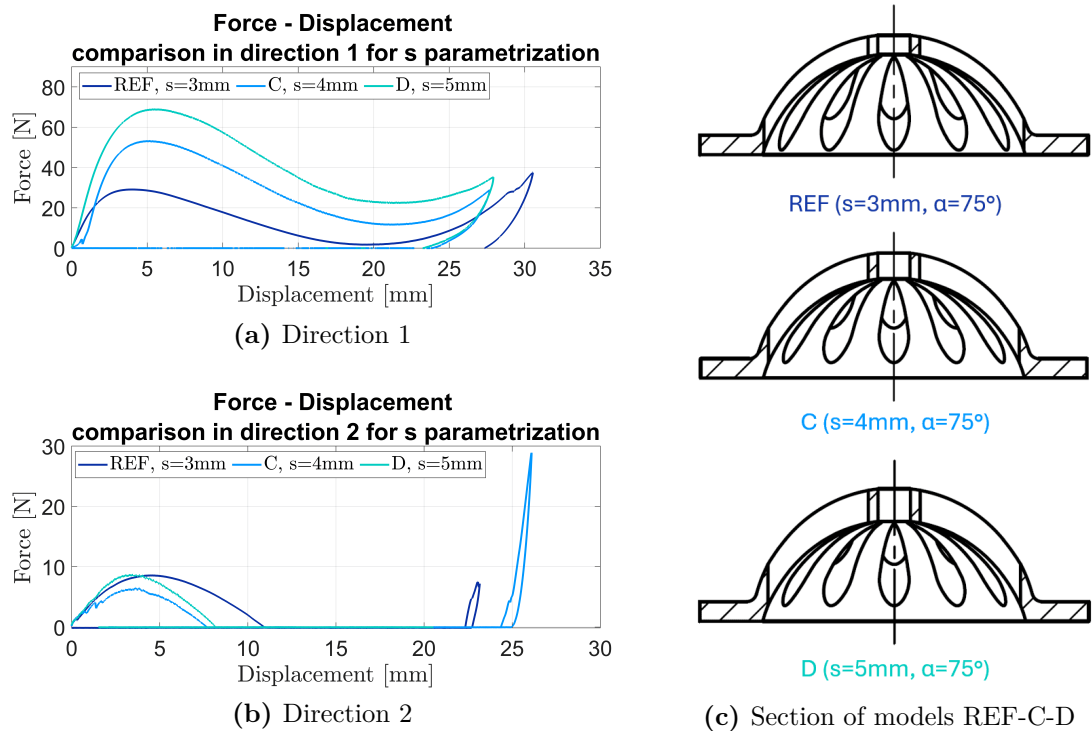
(b) Model B

**Figure 2.11:** Experimental data of model A and model B during compression in direction 1.

### Thickness $s$ parameterisation

In figure 2.12 there is the comparison between model REF, C, D. They have the same angle  $\alpha = 75^\circ$  but with different thicknesses from 3 to 5 mm. The graph for direction 1 illustrates that as the thickness increases, the model shows greater resistance to displacement, as indicated by higher forces at comparable displacements, yet the total displacement remains largely unchanged. This behaviour is a consequence of the fact that the domes in question exhibit a similar shape, figure 2.12(c). In the direction 2 the required force is pretty the same and it is always lower than in the other direction.

In conclusion the thickness affects the force to snap, which is correlated to the closing force of the gripper, but is not related to the passive behaviour which is proportional to the maximum force required in direction 2. This figure demonstrates the impact of varying the thickness  $s$  of the structure on its force-displacement behaviour. The data indicates that thicker models exhibit greater stiffness. A comparison is made between two different directions, with direction 1 showing higher force values overall. This is likely due to differing material properties or structural responses in these orientations.

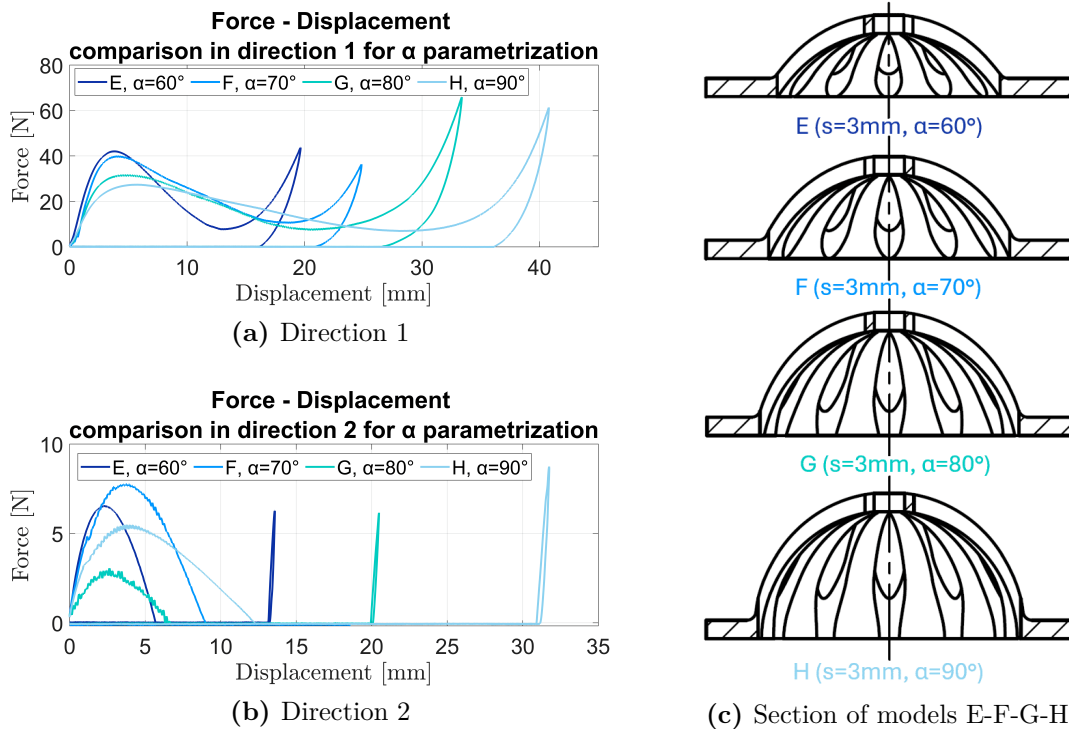


**Figure 2.12:** Comparison of experimental data for the parameterisation of the thickness  $s$  during compression test. In (c) there are the sketch of the 3 models compared.

## Angle $\alpha$ parameterisation

This section compares the models with different angles, which have in common a thickness of 3 mm and angles that increase from  $60^\circ$  to  $90^\circ$ . This comparison is more interesting than the previous one because the shapes of the elements are quite different. This tendency is also illustrated by the curves in the figure 2.13, since they are not only different in terms of a scaling factor, but also in terms of shape. As the angle  $\alpha$  increases, also the total displacement increases and this behaviour is due to geometrical reasons.

With regard to the behaviour in direction 2, it can be observed that the trend is markedly different in terms of maximum force. Model G is able to ensure optimal behaviour in order to obtain a passive closing gripper, as the maximum force is approximately 2 N, despite the order of magnitude being similar to that of the other models analysed. The discrepancy is due to the shape of the bistable element in the snapped configuration. The angle  $\alpha$  affects the shape of the force-displacement behaviour and the total vertical displacement but, in terms of changing the maximum force, it does not play the key role. Combining these results with the previous ones and tuning  $\alpha$  and  $s$ , it is possible to arrange the curve to desired values and shapes of force-displacement characteristic.

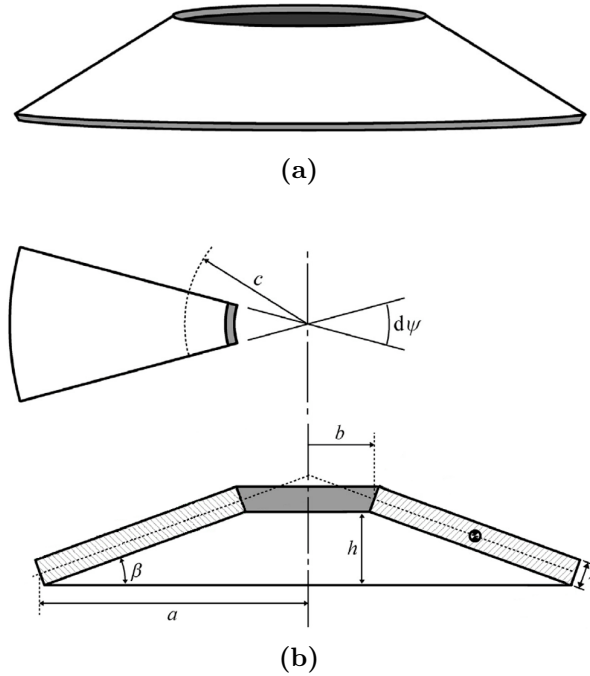


**Figure 2.13:** Comparison of experimental data for the parameterisation of the angle  $\alpha$  during compression test. In (c) there are the sketch of the 4 models compared.

In the next sections, two models are developed in order to simulate the behaviour of this bistable elements in terms of force-displacements trend in order to validate the design with mathematical models and then simulate different shapes not printed.

### 2.2.4 Cup spring model

The first model for the bistable element developed is a mechanical and well known component which is the cup spring. In literature there are a lot of studies about this topic and it has been taken into account because the tendency of the elements reminds the one of these springs and the shape is quite similar. For this thesis the research [28] is considered in order to make comparison between this model and the parameterisation of bistable elements.



**Figure 2.14:** In (a) scheme of the cup spring. In (b) the study model. Image taken from [28].

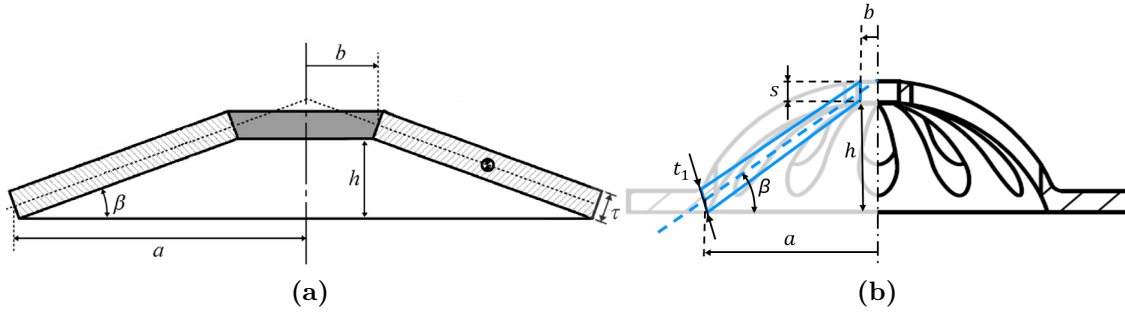
Firstly, with the moment equilibrium for a single spring, the Almen–Laszlo relationship is formulated in equation 2.1 [29]. It allows to correlate the load with respect to the displacement/deflection of the spring.  $E$  is Young’s Modulus,  $\nu$  is the Poisson number,  $\delta$  is the vertical displacement and  $\alpha$  is the radii ratio defined as  $\alpha = a/b$ .

$$P_{L1}(\delta) = \frac{E\delta\pi}{(1-\nu^2)a^2} \left(\frac{\alpha}{\alpha-1}\right)^2 \left[ \tau(h-\delta) \left(h - \frac{\delta}{2}\right) \left(\frac{\alpha+1}{\alpha-1} - \frac{2}{\ln \alpha}\right) + \frac{\tau^3 \ln \alpha}{6} \right] \quad (2.1)$$

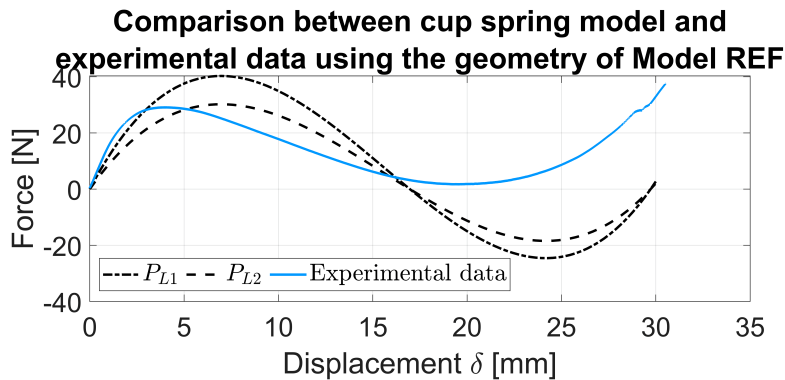
A simplified version of Almen-Laszlo expression was the one presented by Curti, basically it does not consider the Poisson number and it is presented in equation 2.2.

$$P_{L2}(\delta) = \frac{E\delta\pi}{a^2} \left(\frac{\alpha}{\alpha-1}\right)^2 \left[ \tau(h-\delta) \left(h - \frac{\delta}{2}\right) \left(\frac{\alpha+1}{\alpha-1} - \frac{2}{\ln\alpha}\right) + \frac{\tau^3 \ln\alpha}{6} \right] \quad (2.2)$$

In order to make a comparison between this model and the experimental data, the force-displacement graph of model REF in direction 1 is considered. In figure 2.15 is showed the geometric correspondence between the model and the real bistable element. The geometric parameters used are  $a = 20.66$  mm,  $b = 2.5$  mm,  $h = 15.57$ ,  $\alpha = a/b = 8.26$ ,  $s = 3$  mm,  $t_1 = 3.09$  mm. Since the thickness is not constant in the bistable element, but is constant for the model, the parameter  $\tau$  is evaluated as the mean value between the 2 thicknesses  $s$  and  $t_1$ ,  $\tau = (s + t_1)/2$ . The material parameters are taken from the data-sheet of the TPU NinjaFlex Filament 85A by Ninjatek:  $E = 4$  MPa,  $\nu = 0.5$ .

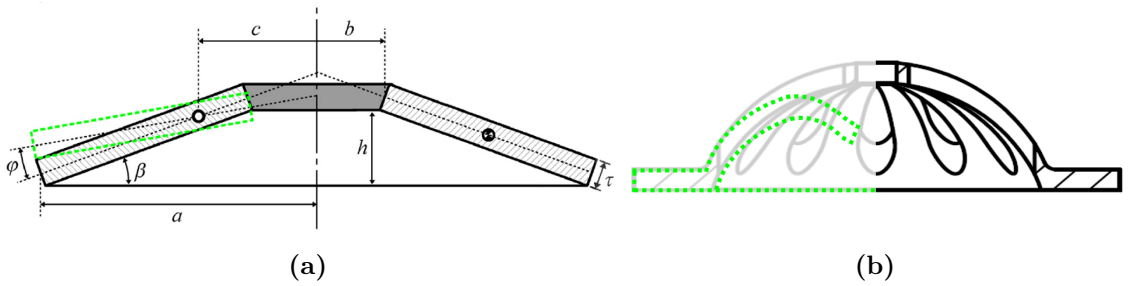


**Figure 2.15:** In (a) cup spring model. In (b) the equivalent for the bistable element where in black there is the real geometry while in blue the approximation to the cup spring model.



**Figure 2.16:** Cup spring model using Almen-Laszlo expression ( $P_{L1}$ ) and Curti's one ( $P_{L2}$ ) and comparison with experimental data of Model REF.

In figure 2.16 is presented the comparison between the experimental data and the two mathematical formulations cited. For small displacement the model can perform quite similar to the experimental data in terms of order of magnitude of required force. For higher displacement the model can not follow the experimental tendency anymore but this is probably due to the shape that the bistable element reaches under load. In figure 2.17 the differences in terms of configuration between the mathematical cup spring model and the bistable compliant element are showed. This behaviour is the reason why the model does not fit the experimental data as the displacement increases because the geometry and type of deformation are completely different.



**Figure 2.17:** Comparison between the deformation to highlight why the cup spring model is no longer valid after a certain value of deformation.

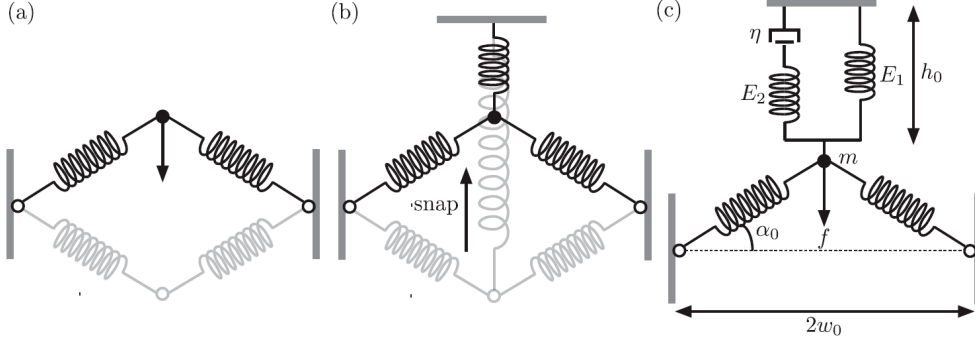
In conclusion the cup spring model is an easy formulation that can be useful to understand the order of magnitude of the stiffness of the bistable element but it is not suitable to find the snapping point between the configurations and simulate different geometries. For this reason another model is implemented in the next section.

## 2.2.5 Compliant snap model

In the literature [27] a snap model with viscoelastic characteristics is adopted in order to find a mathematical relationship between the force and displacement. In the research the element can snap from a natural shape to an inverted one as the element designed for this thesis in figure 2.5.

The model is showed in figure 2.18 and consists on a point mass  $m$  hold by 2 linear spring from the bottom with  $k$  stiffness and a viscoelastic element from the top. This element is a standard linear solid (SLS) consisting of a spring with  $E_1$  stiffness in parallel with a damper  $\eta$  which is in series with another spring  $E_2$ .  $\alpha_0$  and  $\omega_0$  are geometric distances in the rest configuration of the model, representing respectively the angle above the horizontal line for the mass point  $m$  and half the distance between the attachment points on the base. For each bistable model they have been evaluated from the sketch and they are showed in table 2.2.  $f$  is the force and  $h_0$  represents the length of  $E_1$  spring at rest.





**Figure 2.18:** Snapping model. (a) The simplest form of the Mises truss, which features bistable ‘natural’ (highlighted) and ‘inverted’ (lightly shaded) equilibrium states. (b) This bistability is lost when an additional, linearly elastic, spring of sufficient stiffness is attached vertically to the point mass. (c) Replacing the vertical spring by a viscoelastic element, modelled as a standard linear solid (SLS), maintains the bistable–monostable behaviour. Image taken from [27].

In the equations 2.3 the relationship for the non-dimensional force and displacement are showed. The capital letters represents the non-dimensional variables whereas the lower case ones are the real force and displacement.  $f_g$  is the geometric factor, for this thesis is equal to 0.5.

$$\begin{aligned} x &= f_g \alpha_0 w_0 X, \\ f &= k \alpha_0^3 w_0 F. \end{aligned} \quad (2.3)$$

$\lambda$  is defined in equation 2.4 and can be interpreted as the relative stiffness of the upper SLS element compared to the central springs. In the research its range starts from 0 and goes up to 1.5, changing the behaviour of the force-displacement trend. In order to model the bistable element designed, it has been tuned to 0.35 for all the cases.

$$\lambda = \frac{AE_1}{kh_0\alpha_0^2} \quad (2.4)$$

The relationship to evaluate the non-dimensional trend is:

$$F_{\text{eq}}(X; \lambda) = -X(X - 2)(1 - X) + \lambda X \quad (2.5)$$

The method used to evaluate the force-displacement behaviour and to compare the mathematical data with the experimental ones is as follows:

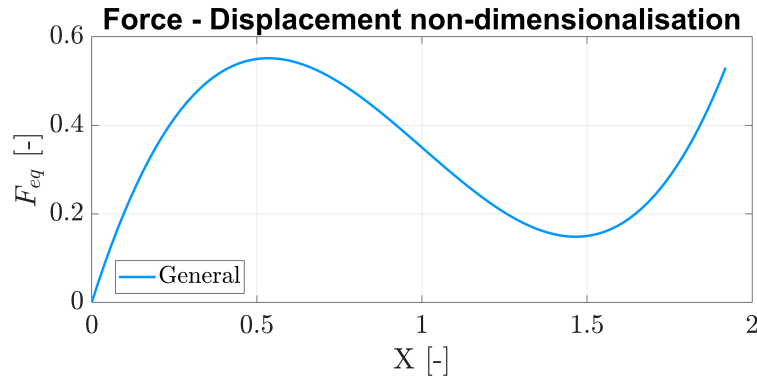
- Find a range for  $x$  displacement from the experimental data
- Evaluate the non-dimensional displacement  $X$  using the relationship 2.3

- Evaluate the non-dimensional force using 2.5 and plot the non-dimensional trend in figure 2.19
- Evaluate the effective force-displacement trend using the relationships in 2.3 and tune the stiffness  $k$  in order to fit the experimental data with the mathematical model. The numerical values of the stiffnesses  $k$  are showed in table 2.2. From the real component is very hard to estimate this parameter and this is why it has been evaluated with this fitting method. The results are quite reasonable because as thickness  $s$  increases, also stiffness  $k$  increases for model REF, C and D. For the others elements the stiffness  $k$  decreases as the angle  $\alpha_0$  increases. This behaviour is coherent with the trend of experimental data showed in figure 2.13

MODEL	$s$ [mm]	$\alpha_0$ [°]	$w_0$ [mm]	$k$ [N/mm]
REF	3.0	75.0	20.7	1.1
C	4.0	75.0	20.8	2.0
D	5.0	75.0	20.9	2.7
E	3.0	60.0	18.5	3.8
F	3.0	70.0	20.1	2.0
G	3.0	80.0	21.1	1.0
H	3.0	90.0	21.5	0.6

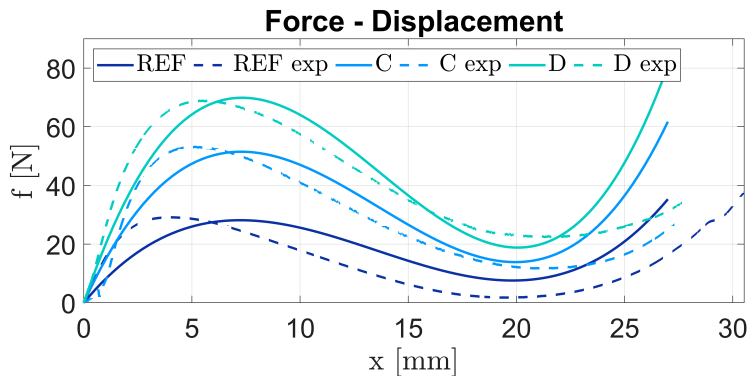
**Table 2.2:** parameterisation of bistable elements for the snap model.

For the models REF, C, D the non-dimensional behaviour displayed in figure 2.19 is the same because the models differ in thickness  $s$  but the shape of bistable elements is the same 2.12(c). The geometric angle  $\alpha_0$  is the same and the distance between the joints at the base is pretty the same, as shown in tab2.2. The only parameter that change is the stiffness  $k$  which does not affect the non-dimensional trend.



**Figure 2.19:** Snap model non dimensional using  $s$  parameterisation.

The non-dimensional approach to the problem is beneficial as it permits the examination of the force-displacement trend with respect to a geometric shape on a graph, independent of the scaling factor. Once the model has been validated with the experimental data, it allows the trend to be predicted for different parameterisation of the bistable element design in order to achieve the desired values in terms of force-displacement.



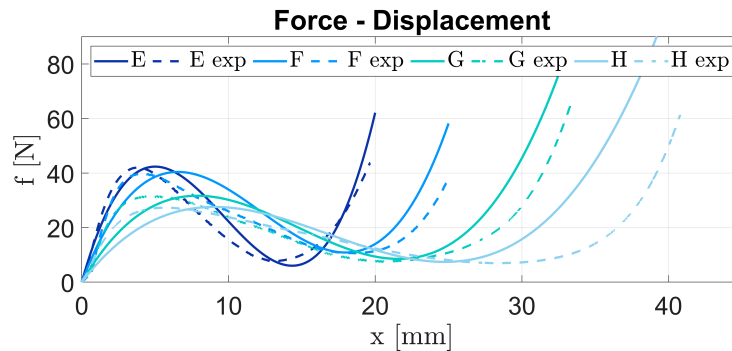
**Figure 2.20:** Snap model using  $s$  parameterisation. Comparison between model and experimental data.

The figure 2.20 presents a comparison between the experimental data and the mathematical evaluation of models REF, C and D in terms of force-displacement. The mathematical model demonstrates a high degree of correlation with the experimental data. In order to perform a numerical evaluation of the error, two points are considered: the maximum force and the snapping point, which is characterised by the local minimum of the curve in terms of maximum force with a general delay for the displacement. The numerical evaluation is presented in tables 2.3 and 2.4. Considering the first 3 models, for the fitting of the maximum force point the normalised error is higher in terms of displacement because it goes from 7 to 13% and the error of the force required is around 3%. The table 2.4 shows the error of the snapping point between the model and the experimental data in terms of displacement and force. In this case the error for the force is quite higher than in the previous case, the order of magnitude is around 4-5%. The error of the REF model is 21% and it is very high compared to the other cases, a possible cause is the fact that this model was printed and used more, so the resistance of the 3D printed element could be slightly compromised. The discrepancy in the results may be attributed to the fact that the initial three models were examined with identical parameters for the non-dimensional curve. However, it is plausible that the tested elements may exhibit slight variations in their geometrical characteristics with respect to one another, due to the inherent limitations of the 3D printing technique.

The occurrence of this error can be reduced by adjusting the value of the  $k$  stiffness

parameter, which represents a gain for the amplitude of the curve. For this thesis, the aim is to adapt to the global trend; it is possible to perfectly adapt the snapping point by adjusting the parameter  $k$ , but the error of the maximum force would increase greatly and the global behaviour of the curve would no longer follow the experimental trend.

The non-dimensional graphs for the  $\alpha$  parameterisation, which means the comparison between the models E, F, G, H is not presented because for each model there is a different graph depending on the geometric shape that, in this case, changes every time as shown in figure 2.13(c).



**Figure 2.21:** Snap model using  $\alpha$  parameterisation. Comparison between model and experimental data.

In figure 2.21 it is showed the comparison between models E, F, G and H that differ from each other for the geometric angle  $\alpha$ . Also in this case it is possible to evaluate the error between the maximum force point and the snapping point. The table 2.3 shows the error in the maximum force point and the error in the force is impressive because in the worst case it is 2% and it is quite smaller than the error in the displacement which goes from 6 to 9%.

In the end the snap compliant model adopted is quite robust and valid to model the bistable element developed and more suitable than the cup spring model of the previous section. For the tests carried out and focusing on the 2 points studied that characterise the curve, this model works approximately well with a percentage error of 7% for displacement and 3% for force, so it is suitable for predicting the order of magnitude of a new model with different parameterisation, if necessary. In order to better validate this model, all the bistable elements can be reprinted with different materials, tuning the stiffness  $k$  each time, so that this method can be verified for the same geometry but with different parameters.

For the bistable grippers developed in the next section, the model REF has been used because is the best trade-off for this application in terms of energy-required to make it snap between the two configurations and total displacement between them.

Maximum force point								
	Experimental		Snap model		Error		Normalised error	
Model	$x$ [mm]	$f$ [N]	$x$ [mm]	$f$ [N]	$\Delta x$ [mm]	$\Delta f$ [N]	$\Delta x/x_{max}$	$\Delta f/f_{max}$
REF	4.0	29.1	7.4	28.1	3.4	1.0	13%	3%
C	5.0	53.1	7.4	51.5	2.4	1.6	9%	3%
D	5.4	68.8	7.4	70.0	2.0	1.2	7%	2%
E	3.8	42.0	4.9	42.3	1.1	0.3	6%	1%
F	4.1	39.8	6.3	40.4	2.2	0.6	9%	2%
G	5.1	31.5	7.7	31.7	2.6	0.2	8%	1%
H	5.8	27.3	9.1	27.6	3.3	0.3	8%	1%

**Table 2.3:** Analysis of maximum force point. Comparison between snap model and experimental data.

Snapping point								
	Experimental		Snap model		Error		Normalised error	
Model	$x$ [mm]	$f$ [N]	$x$ [mm]	$f$ [N]	$\Delta x$ [mm]	$\Delta f$ [N]	$\Delta x/x_{max}$	$\Delta f/f_{max}$
REF	19.2	1.8	19.9	7.9	0.7	6.1	3%	21%
C	21.4	11.7	19.9	13.8	1.5	2.1	6%	4%
D	21.4	22.5	19.9	18.8	1.5	3.7	6%	5%
E	13.0	7.7	14.3	6	1.3	1.7	7%	4%
F	18.7	10.7	17.9	10.9	0.8	0.2	3%	1%
G	20.8	7.7	21.7	8.5	0.9	0.8	3%	3%
H	28.4	6.9	24.9	7.4	3.5	0.5	9%	2%

**Table 2.4:** Analysis of snapping point. Comparison between snap model and experimental data.

## 2.3 Gripper design with bistable element

This section is concerned with on the design and development of a compliant and bistable gripper for a soft robotic arm. The compliance of the gripper allows it to interact safely with fragile objects, reducing the risk of damage, while its bistable mechanism enables reliable and energy-efficient gripping and release operations. The bistability ensures that the gripper can maintain its grip without continuous power input, which is critical for energy efficiency and autonomous operation of soft robotic systems.

The design aims to address the challenges of creating a gripper that is both adaptable to a wide range of objects and capable of maintaining a stable grasp under varying conditions. By integrating the compliant bistable element developed in 2.2, the proposed gripper design provides a versatile and robust solution that enhances the

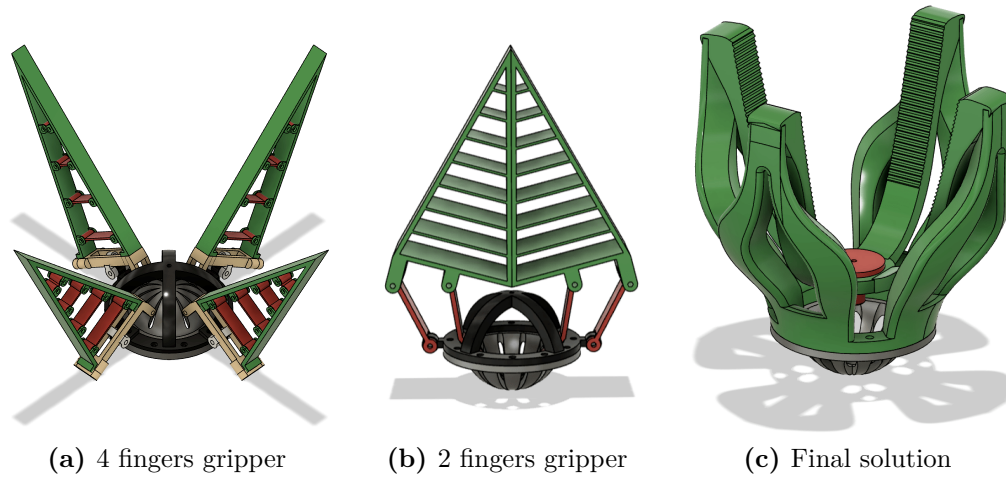
functionality of the soft arm, making it suitable for complex, real-world tasks. The following sections present a detailed account of the specific design considerations, material selection, mechanisms employed and anticipated performance outcomes of the compliant and bistable gripper.

### 2.3.1 Overview of solutions

A lot of solutions have been explored in order to find the right and more suitable design for the gripper. The figure 2.22(a) shows a solution with 4 compliant fingers attached to the bistable element by means of the orange supports. Upon the bistable element snap, the inner portion of the orange supports exhibits a displacement in accordance with the deformable movement of the dome. This results in a rotation of the triangular fingers around the white element, which behaves in a manner analogous to a pivot and ensures the closure of the fingers. To control the gripper is mandatory to snap the bistable element which is actuated by means of 2 tendons in opposite directions and assembled on a pulley of a rotary motor. In both cases the tendon is connected to the bistable element at the central tip of the dome. The principal issues associated with this solution are related to the fact that the bistable element in its rest position corresponds to the gripper in its open configuration. Consequently, when the gripper is in the closed position, the bistable element is forced into a snap position, which does not allow for the maximum resistance force that could be achieved in the natural configuration. Another problem of the 4 fingers solution is the fact that in the closed position of the gripper there is a gap of around  $1\text{ cm}^2$  that is not suitable for very small objects and the TPU fingers are too stiff to deform easily according to the shape of the object to grasp.

The 2 finger solution shown in figure 2.22(b) has been developed in order to solve some problems of the previous version. First of all the bistable element is mounted in the opposite direction in order to be in the natural configuration when the gripper is closed and ensure an higher closing force on the object to grasp. This design allows also to work with really small objects as the fingers are in contact when the gripper is closed. The triangular fingers are connected to the bistable element by means of two red rigid links and they operate according to the same working principle as the four-finger solution. Also for this solution the actuation of the dome is due to two antagonistic tendons that acts directly on the bistable element. The rods that comprise the triangular shape are flexible and are manufactured as a single component with the finger itself in order to reduce the stiffness and adapt better to objects. The primary issue with this design is that the closing behaviour of the gripper is constrained to a single plane but, since the twist of the soft arm is not allowed, is impossible to control and adjust the rotation of the gripper. Furthermore, both solutions present mechanical issues due to the necessity of assembling numerous components via small screws. Therefore, it is not possible to regulate and restrict the occurrence of friction.

The figure 2.22(c) presents the final design of the bistable and compliant gripper that is discussed in the next section.

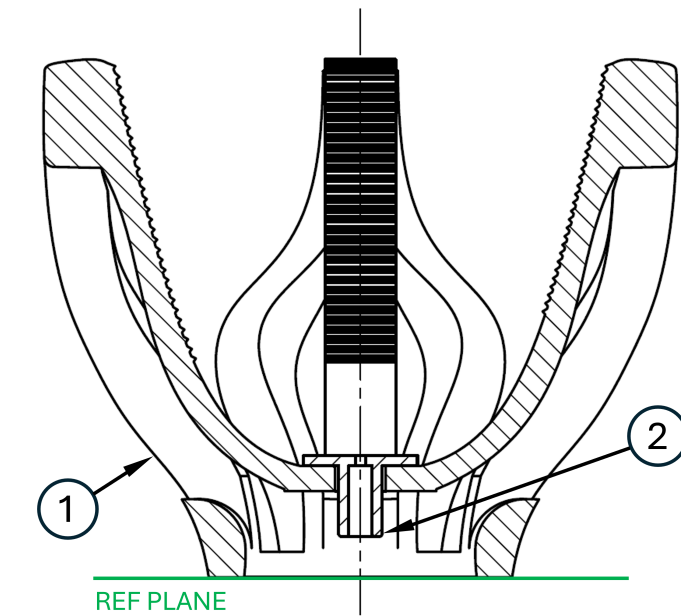


**Figure 2.22:** Overview of gripper designs. The red and orange components are rigid PLA elements that make up mechanisms and transmission organs. The black PLA component is part of the support structure that allows to hold the tendon in order to actuate the bistable element in both directions. In grey there is the bistable element whereas the compliant part is coloured in green. Both of them are printed in TPU.

### 2.3.2 Final solution of bistable gripper

The final design of the bistable gripper is presented in figure 2.22(c). This solution represents an optimal balance between compliance, closing force, and versatility. To ensure the maximum closing force of the gripper, the bistable element is in the natural pose when the gripper is closed (in the figure, both the gripper and the bistable element are presented in their rest configuration, as they have been 3D modelled. However, they are not connected in this figure). Compared to other solutions all the closing mechanism is innovative because the actuation tendon is no more connected directly to the bistable element but a new system has been developed and it will be discussed later. There are no more classical triangular fingers and the design is bio-inspired by flowers. The principal advantage of this solution in comparison to previous ones is that the entire gripper is constructed from a single element, thus eliminating the possibility of friction between components that are not ideally aligned and resulting in the generation of an undesired resistance. In the open configuration, the gripper is capable of inscribing a circle with a diameter of 90 cm, which serves to determine the maximum size of objects that can be gripped. To describe better the system, a section of the gripper is presented in figure 2.23.

Many attempts with different materials and printing parameters has been done to make comparisons but the final gripper (element 1) has been 3D-printed in TPU with NinjaFlex Filament 85A by Ninjatek with these features: Young Modulus  $E=4$  MPa, ultimate elongation 660%. The print parameters used for the final solution are  $T_{nozzle}=235^{\circ}$ ,  $T_{bed}=30^{\circ}$ , layer height=0.10 mm, print speed=30 mm/s, infill=20%. These parameters ensure high compliance with the structural requirements and guarantee lightweight construction, which is a crucial feature for this application given the payload limitations of the soft arm. The configuration of the gripper is closely associated with the dome shape, as the cap (component 2) is affixed to it. Consequently, the vertical displacement of the bistable element activates the gripper. The fingers are slightly indented in order to facilitate a more secure grasp of the object to be held, thereby ensuring a more stable and secure hold.



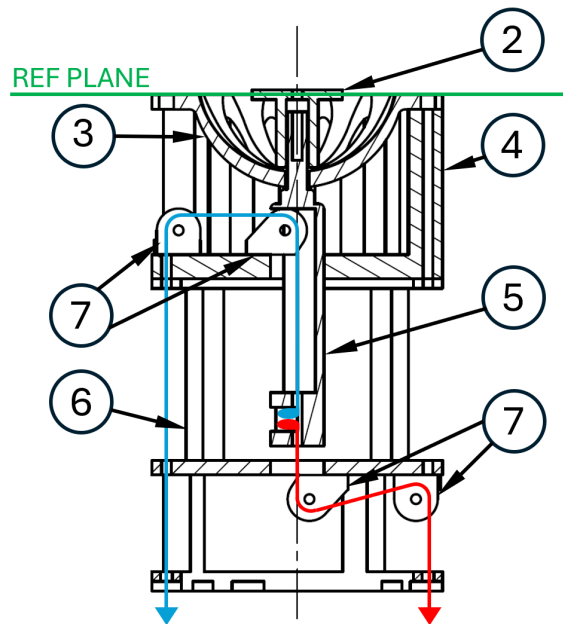
**Figure 2.23:** Section of bistable gripper. (1) is the compliant gripper itself, (2) is the cap that allows to make the gripper integral to the bistable element.

The actuation mechanism is illustrated in figure 2.24 with the bistable element in the rest configuration with the cap under the REF plane. In contrast, in figure 2.23 the cap is above the green plane, which indicates that the bistable element should be in the snapped configuration. The green REF plane is displayed to set a global reference system between the two drawings.

The actuation tendons, displayed in red and blue, traverse the entire soft robotic arm and are connected to a motor by means of a pulley. Unlike the systems in previous cases, the tendons are not directly connected to the bistable element but they are both attached to the actuation rod (element 5). This component is affixed to the cap



via a small screw that enters from the top and secures the two elements together with the bistable element situated in the middle. In accordance with the configuration illustrated, actuation of the blue tendon results in upward movement of the rod, which causes the bistable element to snap and push the cap above the reference plane. If the gripper is attached to the element 2 as shown in figure 2.23, this behaviour correspond to opening movement of the gripper. Conversely, in order to close the gripper, the red tendon must be actuated and returned to the configuration depicted. The cap returns to its original position above the REF plane, and the fingers move to the central axis, in this way they close the gripper system. Once the gripper is in a closed position, the red tendon can be released, as the resistance of the bistable element ensures that the closing force on the object is maintained without the need for motor actuation.



**Figure 2.24:** Actuation of bistable element in order to control the gripper. The sketch represents the close position. In red there is the tendon that control the closure of the gripper while the tendon in blu is the one the allows to open it. (2) the cap already shown in 2.23, (3) bistable element, (4) bistable element support, (5) actuation rod, (6) roll bearings support, (7) roll bearings.

The principal issue for this solution is that the stiffness of the bistable element need to work with the one of the gripper itself, it means that if the the bistable element is not stiff enough, the gripper may be unable to maintain a closed position and grasp an object. Vice versa if it is too stiff, the required force to snap it back increases greatly, which makes it more difficult to passively close the gripper. The passive behaviour is characterised by the application of a force to an object by the robotic

arm, which generates a force on the cap that causes the bistable element to snap back and, consequently, close the gripper and grasp the object. This configuration permits the desired behaviour to be achieved, but only for a specific set of objects to be grasped. The objects must be in direct contact with the cap in order for the required force to be generated; contact with the finger alone is insufficient. The same considerations can be applied to the stiffness of the fingers. If the fingers are too flexible, the resistance of the bistable element may be insufficient to close them. Conversely, if the fingers lack sufficient stiffness, the closing force on the object to grasp may be inadequate.

Since the vertical displacement of the rod is closely related to the shape of the gripper in the closed position, the design of the gripper itself depends on it. With a total displacement of the bistable elements that start from 20 mm and goes to 40 mm depending on the model, the working range is limited because in the 2 snapped configuration the gripper should be open and closed with a fixed vertical displacement. For this reason the distance between the fingers in the open position is 90 cm and can not be larger.

After testing all the models discussed in chapter 2.2 the REF model is the most suitable for this application. The total displacement of 25 mm ensures a secure closure of the gripper, with a relatively low actuation force and sufficient stiffness to provide a robust closing force. Consequently, the REF model represents an optimal balance between these competing considerations. With this solution the gripper in closed position presents a gap of  $2\text{ cm}^2$  between the fingertips that make the gripper not suitable for very small objects.

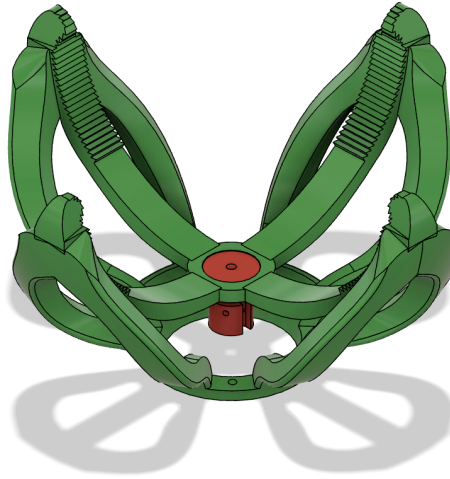
## 2.4 Final design

The pursuit of optimisation of soft robotic systems continues to stimulate innovation in gripper technology, particularly in response to the constraints of preceding designs. Although the earlier bistable gripper design offered considerable advantages in terms of energy efficiency and the ability to maintain a stable grip, certain limitations have emerged that restrict its effectiveness in real-world applications. These limitations include restricted payload capacity, difficulties in handling objects of varying sizes, and challenges in adapting to a wider range of practical tasks.

This chapter proposes a new soft gripper design to address these issues. Departing from the bistable mechanism, the revised design focuses on increasing the payload capacity of the gripper, improving its adaptability to different object sizes, and making it more suitable for a wider range of real-world applications. The compliance of the gripper remains a key feature, ensuring that it can still safely and effectively interact with fragile and irregularly shaped objects, while the design changes provide greater flexibility and control during gripping operations.

The following sections describe the design, the modifications made to the gripper

structure, and the expected performance improvements. This new approach aims to provide a more versatile and capable gripper that can better meet the demands of different applications, further advancing the integration of soft robotic systems into practical use cases.

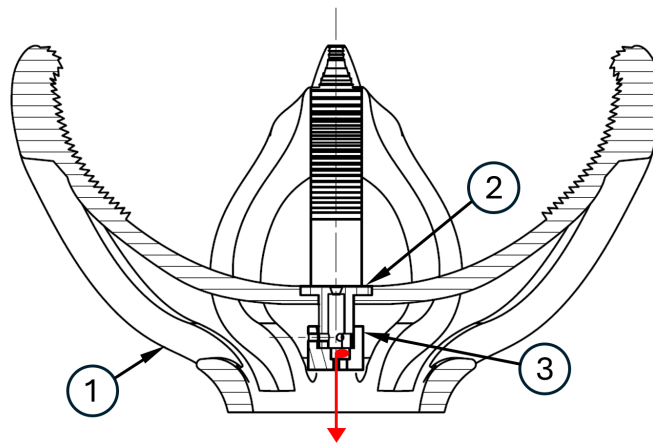


**Figure 2.25:** Model of the gripper without bistable solution. In red the PLA components that make up the transmission organ and in green the flexible gripper.

The figure 2.25 illustrates the design of the final gripper which is quite similar to the bistable one. Following the removal of the vertical displacement constraint, a minor redesign of the gripper has been implemented. This has resulted in a wider open position and an increased distance between the fingertips, reaching 150 cm from the previous 90 cm observed in the bistable model. For this reason the size of the objects that can be grasped has increased. Modelling the tips with a tapered shape, this new solution is also able to perform with really small objects because there is no more gap between the fingers in the closed configuration. Additionally, the external configuration of the gripper has undergone a slight modification in comparison to the previous version. This alteration has been implemented with the objective of facilitating the formation of a secure grip on objects, while simultaneously preventing the formation of significant gaps between the fingers.

The section of the model is showed in figure 2.26 which demonstrates the mechanism for closing the gripper. The cap (element 2) is positioned into the soft gripper by means of a geometric interlocking and it has an internal hole where a small camera can be housed enabling the utilisation of another point of view as an additional functional element when the object to be grasped is not visible from the user. The cap has 3 radial holes necessary to fix this component to the tendon attachment support (element 3) with three M2 screws. The actuation tendon is attached to this last element that, upon activation, moves downwards causing the gripper to

close to the central axis line. Given that this solution is no longer bistable, the gripper maintains the closed position until the motor maintains the tendon in an actuated state. When the tendon is released, the compliant structure manage to recover its original open shape without extra mechanisms. The final gripper has been 3D-printed in TPU with NinjaFlex Filament 85A by Ninjatek with these features: Young Modulus  $E=4$  MPa, ultimate elongation 660%. The print parameters used for the final solution are  $T_{nozzle}=235^\circ$ ,  $T_{bed}=30^\circ$ , layer height=0.10 mm, print speed=25 mm/s, infill=70%. In essence, the printing parameters remain consistent with the bistable model. However, a notable distinction is the infill setting, which is adjusted to 70% to enhance the stiffness of the fingers and improve the closing force.



**Figure 2.26:** Section of the gripper without bistable solution. (1) is the compliant gripper, (2) is the cap that allows to actuate the gripper (3) is the tendon attachment support. In red it is represented the tendon that control the closing behaviour.

## 2.5 Comparison with bistable solution

This section presents a quantitative comparison between the bistable and monostable gripper designs, focusing on key performance metrics such as range of motion, object adaptability, energy efficiency, and force output, with particular consideration of the weight of the objects to be grasped. The aim is to provide a comprehensive understanding of the trade-offs associated with each design. Due to the payload limitations of the arm, it is important to note that the weight of the system, including the bistable gripper and actuation, is approximately 114.2 g, while that of the final gripper is 127.4 g. Both solutions have been designed with the objective of achieving the lowest possible weight. The bistable gripper is lighter than the final one because it is smaller and the infill ratio is 20% instead of 70% of the final solution but all the more complex system of actuation is heavier and the total amount are almost identical.

### 2.5.1 Experimental setup

To ensure the accuracy of the payload capacity of the gripper, a controlled testing procedure is implemented to eliminate any potential external variables that could arise from the soft robotic arm itself. The robotic arm is placed in a stationary, vertical position throughout the test to eliminate any potential influence from arm movement. The gripper is then required to grasp a series of objects, each with varying weights and sizes, to assess its capability across potential use cases.

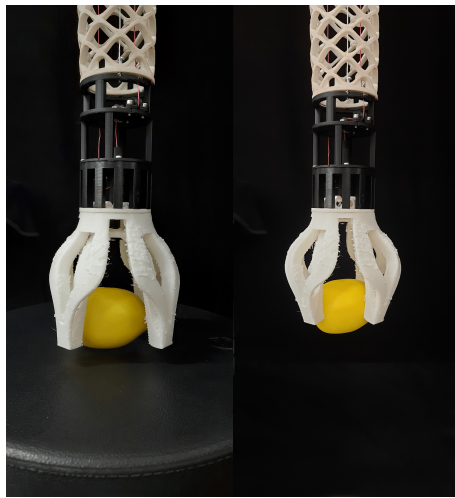
The objects are positioned on an adjustable-height plane, allowing for precise control over the initial grasping conditions. By adjusting the height of the plane, the test simulates different grasping scenarios, ensuring that the performance of the gripper is rigorously evaluated under consistent and repeatable conditions. This approach not only isolates the payload capacity of the grippers from the dynamics of the robotic arm but also provides a reliable framework for comparing the performance of different gripper designs in handling various objects.

### 2.5.2 Validation

In order to validate and make quantitative comparisons between the two main solutions developed, a set of objects with varying sizes, geometries and weights has been taken into account. This approach serves not only to validate the functionality of the grippers, but also to facilitate a detailed comparison between the bistable and monostable models. The surface quality of the object is another factor that affects the outcome of the grasping test. However, this factor is not taken into account due to the difficulty in accurately assessing it. The results are showed in table 2.5.

Object	width depth height [mm]	weight [g]	Bistable	Final
Tomatoes	40 x 40 x 20	35.6	Yes	Yes
Strawberries	30 x 10 x 10	19.6	No	Yes
Lemon	70 x 50 x 50	85.3	Yes	Yes
Water bottle	70 x 70 x 220	597.8	No	Yes
Screwdriver	15 x 15 x 125	26.5	No	Yes
Screw	3 x 3 x 30	2.0	No	Yes
Scotch	132 x 132 x 50	423.3	No	Yes
Pla cube 1	50 x 50 x 50	16.9	Yes	Yes
Pla cube 2	75 x 75 x 75	38.1	Yes	Yes
0.5 kg weight	99 x 99 x 15	500.1	No	Yes
Spacemouse Joystick	77 x 77 x 55	439.6	No	No
Play station Joystick	153 x 59 x 54	216.5	No	Yes

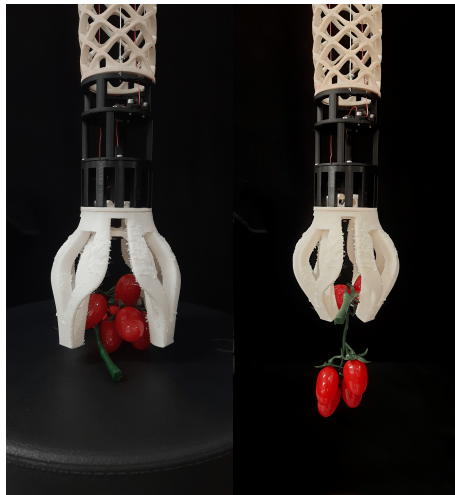
**Table 2.5:** Comparison between the 2 grippers. The dimensions of width, depth and height are estimated approximately.



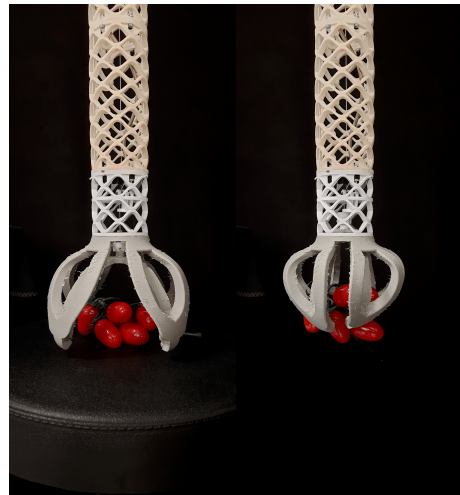
(a) Bistable gripper with lemon



(b) Final gripper with lemon



(c) Bistable gripper with tomatoes



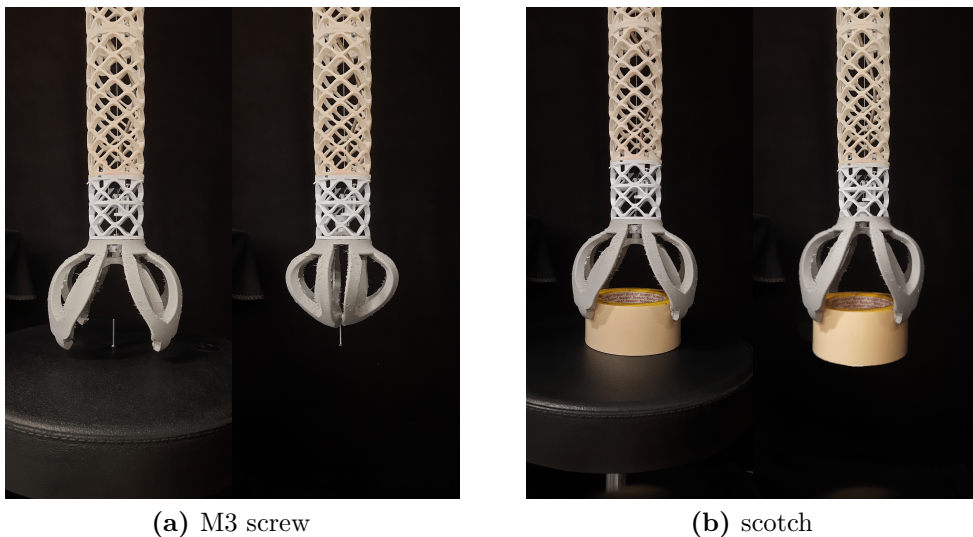
(d) Final gripper with tomatoes

**Figure 2.27:** Compliant behaviour of the grippers and comparison between the 2 solutions. In figure (a) and (c) the bistable gripper while in the figure (b) and (d) the final one.

The results presented in the table 2.5 show how the final design is more suitable with a larger variety of objects in terms of size and weight. The final version is capable of grasping objects that are otherwise impossible to grip for the bistable solution but there are no cases where vice versa is valid. In terms of weight the limit condition for the bistable solution is around 85 g whereas for the final design is up to 600 g. This discrepancy can be attributed to the fact that the closing force is guaranteed by the tendon actuated by the motor, as opposed to the resistance of a compliant bistable element. In terms of the dimensions of the objects, the bistable

solution is capable of accommodating items within a range of 40x40 cm to 75x75 cm. In contrast, the workspace of the final solution is significantly larger, spanning from 3x3 cm to 130x130 cm.

The figure 2.27 illustrates how the two solutions perform with the same objects, respectively a lemon and tomatoes. In both cases the bistable gripper grasp the object using just the tip of the finger, for both objects the passive behaviour is not possible because of the shape. The bistable gripper is less rigid than the final gripper, which allows it to adapt more effectively to the shape of the objects to grasp. On the other hand, the final gripper is capable of applying a greater closing force, which results in a more secure hold on the objects in comparison to the previous model. Finally figure 2.28 illustrates the workspace of the objects that the final gripper can accommodate. It can be considered universal in its workspace with regard to both the size and shape of the objects that it can hold.



**Figure 2.28:** Workspace of the final gripper. In figure (a) there is a M3 screw to grasp to highlight the capability of the gripper to grasp very thin objects. In figure (b) there is a scotch with diameter of 132 mm which is close to the maximum size of the gripper.

In conclusion, the versatility in terms of weight and size of the objects makes the final gripper the optimal trade-off. The advantages of the bistable solution include adaptability to shape and energy saving. However, in practical applications, these advantages are outweighed by those of the final gripper. This solution remains sufficiently adaptable for use with compliant grippers.

# Chapter 3

## Robot control

This chapter presents an analysis and solution to the inverse kinematics (IK) problem for the Helix soft robot. The study begins with the formulation of the forward kinematics (FK), which establishes the relationship between the input parameters and the corresponding position and orientation of the end effector. On this basis, the IK model is then developed in order to determine the necessary inputs required to achieve a desired position and orientation of the end-effector.

Subsequently, the workspace of the robot (WS) is examined, with particular attention paid to the boundary conditions that limit its movement capabilities. This entails the identification of the physical constraints and mechanical limits that define the range of achievable positions and orientations within the operational space.

In consideration of the open-loop control nature of the system, a validation section has been included with the objective of assessing the performance and accuracy of the proposed kinematic models. The intrinsic error of the system is then evaluated by means of comparison between the theoretical trajectories derived from the IK solution and the actual trajectories recorded by an optical tracking system. The error analysis quantifies the discrepancies between the desired and achieved positions, thereby providing insight into the precision and reliability of the configuration of the robot.

The objective of this chapter is to provide a comprehensive understanding of the kinematic behaviour of the soft robot, from theoretical modelling to practical validation.

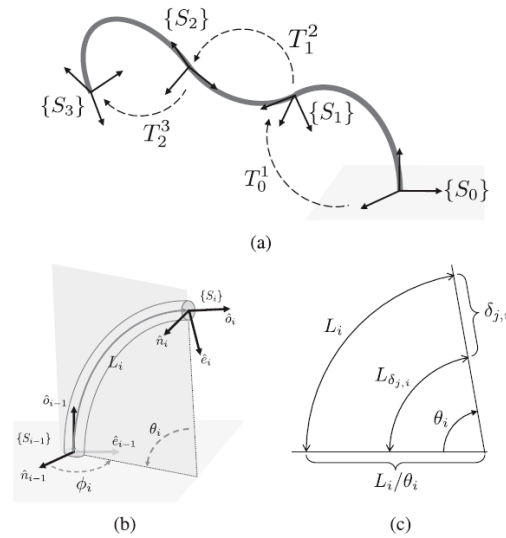
### 3.1 Kinematics and PCC model

In general the kinematics in robotics involves the study of the motion of robot components without considering the forces or torques that cause them. It focuses on the relationship between joint parameters and the position and orientation of the end effector. In the case of traditional robots, rigid joints are typically actuated



by rotary motors. The angular displacement can be evaluated by assessing the position of the rigid body, which is equipped with an additional motor at the tip that controls another body, and so on until the end effector. The combination of the kinematic relationships in three-dimensional space allows the position and orientation of the tip of the robot to be determined. For the Helix arm there are no rigid joints connected to rigid bodies but tendons actuated by rotary motors, the logical flow of the kinematics is pretty similar but with different relationships. Before investing the kinematics, a quick summary of the PCC model is presented.

The robot has been developed under the PCC model which means that each section is approximated to arc with constant curvature. The figure 3.1(a) illustrates the complete configuration of the soft arm, with the central line representing the axis of the manipulator. This axis is split into three arcs, which represent the actual sections. The reference systems are situated at the extremities of each arc with the correct orientation that can be evaluated by means of the rotational matrices. A single section is represented in figure 3.1(b-c) to identify the geometrical dimensions that are typical of the PCC model that is under consideration.



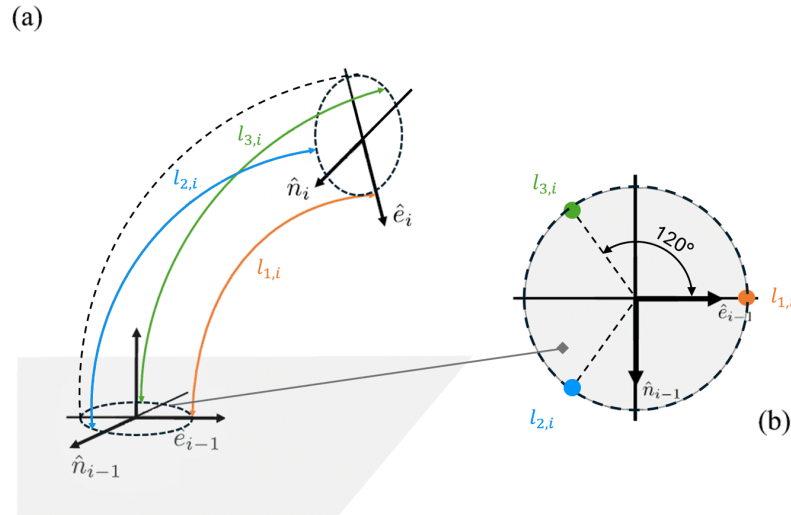
**Figure 3.1:** PCC soft robot sketch. In (a) there are represented 3 sections of a soft robot with the relative rotational matrices and the relative reference systems of each section. In (b) a single section of CC with the geometrical dimensions to describe the model. In (c) a planar sketch to highlight better the quantities. Image taken from [18].

## 3.2 Forward Kinematics

The FK developed comprises an evaluation of the end effector position and orientation, initiated from the actuation stage, which employs three distinct states to define the configuration of the arm. In order to avoid overburdening the mathematical formulae, it should be noted that the kinematics analysis presented in this chapter is limited to the position of the end effector. However, the implementation of the code, which includes both position and orientation, can be found in the appendix A. The following illustrations demonstrate the utilisation of a single section of the robot to enhance the clarity and comprehensibility of the drawings, the subscript ' $i$ ' represents the  $i$ -th section:

- **Tendon lengths:** The input command to actuate the robot is the position of the rotary motor. Given that the tendons are attached to the pulley by a 10 mm radius, it is a relatively straightforward process to convert the angle position of the motor into the linear length of the tendon. This parameter is therefore taken into account as the input command for the kinematics analysis. In each section, three tendons are arranged along a circle with an angular distance of  $120^\circ$  between them as illustrated in figure 3.2.

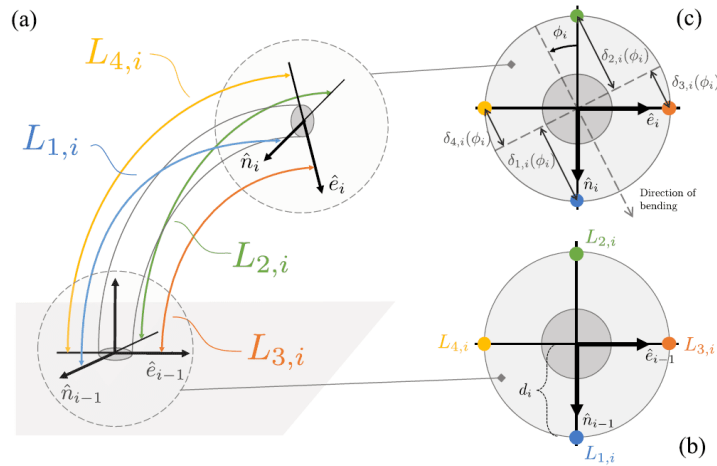
The actuation of the tendons independently permits the attainment of all possible compressed and bent configurations within the limits of the WS. To guarantee that the tendon lengths are aligned with the actual position and the PCC model is accurate, it is essential that all the tendons are maintained in a state of tension at all times.



**Figure 3.2:** Representation of the tendons of one section  $i$ -th. In (a) there is a 3D representation to visualize the section, figure (b) shows the normal point of view of the section  $(i-1)$ -th in order to illustrate how the tendons are arranged.

- Configuration space:** The configuration space, also called q-parameterisation, is the novel PCC modelling developed in [18]. In a manner analogous to the approach taken with rigid robots operating within a defined range of measurable quantities, such as joint angle, this parameterisation involves the utilisation of tangible, quantifiable parameters with the aim of determining the current configuration. The variables taken into account are four arcs for a single section as shown in figure 3.3. These arcs must be arranged with an angular distance of  $90^\circ$ . As a result, it is possible to evaluate the PCC arc configuration with three variables for each section  $q_i = [\Delta_{x,i} \ \Delta_{y,i} \ L_i]$ , which contain the same information as the classical PCC model in figure 3.1 but with a direct connection to measurable quantities. The mathematical relationships are developed in the next section.

The term  $\delta_{j,i}$  represents the distance between the end of the  $j$ -th arc of the section  $i$ -th and the straight line included in the plane  $\hat{n}_i - \hat{e}_i$ , which is perpendicular to the direction of bending and passes through the central axis of the segment.

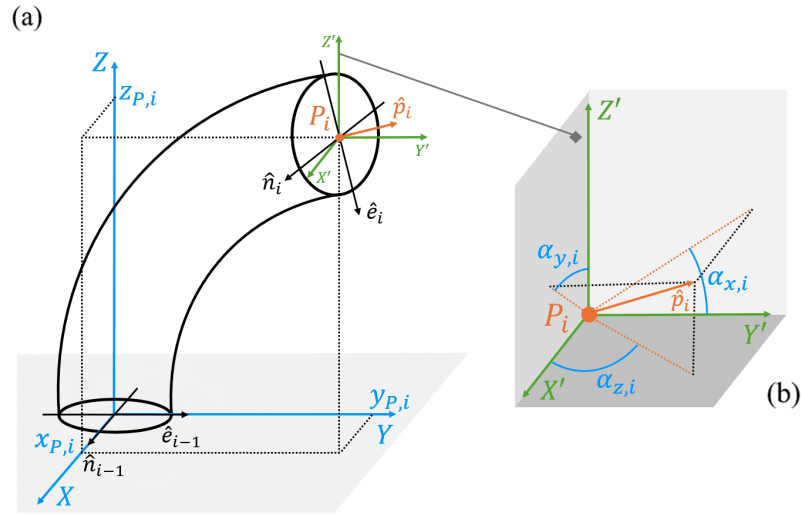


**Figure 3.3:** Representation of the q-parameterisation, for the  $i$ -th segment of a PCC robot. Panel (a) shows the  $i$ -th CC segment. In (b) there is the  $(i - 1)$ -th section where the two ends of each arc are connected at a distance  $d_i$  from the origin of  $S_{i-1}$ . (c) shows the  $i$ -th cross section and the relevant quantities are underlined in the figure. Image taken from [18].

- Cartesian space:** The last space involved to describe the model is the cartesian one. In figure 3.4 are illustrated the quantities that describe the position and the orientation of the tip of the section  $i$ -th. The reference system XYZ represents the global coordinate system, which is fixed to the ground and allows the coordinates of the point  $P_i = [x_{p,i} \ y_{p,i} \ z_{p,i}]$  to be found. This represents the

position of the central point of the last section of the arc  $i$ -th. Combining three sections, this point corresponds to the end effector position.

In order to fully describe the configuration, also the orientation of the vector  $\hat{p}_i$  has to be found. The local reference system  $X'Y'Z'$  is oriented as the previous one but the origin is located in the point  $P_i$ . The orientation of the vector is evaluated by means of three angles  $[\alpha_{x,i} \ \alpha_{y,i} \ \alpha_{z,i}]$  illustrated in figure 3.4(b) which represents the rotation along the three axes. As the reference systems are aligned, the rotations in the local system are the same as in the global system. Using the six variables presented, it is possible to fully evaluate the position and orientation of the end effector in the ground/arm reference system.



**Figure 3.4:** Representation of cartesian space of one section  $i$ -th. The point  $P_i$  passes on the central axis and the vector  $\hat{p}_i$  is normal to the section  $i$ -th. In (a) there is the representation of the 3D section with the translation variables highlighted that allows to determine the position of the point  $P_i$ . In (b) there is the local referene system with the rotational variables that identify the orientation of the tip of the section.

### 3.2.1 From tendon lengths to configuration

This section presents the kinematics relationships used to pass from the tendon lengths to the configuration space. In order to make the analysis as general as possible, the subscript ' $i$ ' stands for the section  $i$ -th. The flow is illustrated in equation 3.1 from the tendon lengths space  $T_i$  to configuration  $q_i$ . In equation 3.2 the function  $G_i$  permits the calculation of the  $i$ -th configuration starting from the  $i$ -th tendon lengths which is the aim of this subsection.  $l_{j,i}$  is the length of the tendon  $j$ -th of the section  $i$ -th,  $\Delta_{x,i}$  and  $\Delta_{y,i}$  are respectively the inclination of the  $i$ -th

section along y and x axis and  $L_i$  is the length of the central axis of the  $i$ -th arc.

$$T_i = [l_{1,i} \quad l_{2,i} \quad l_{3,i}]^T \Rightarrow q_i = [\Delta_{x,i} \quad \Delta_{y,i} \quad L_i]^T \quad (3.1)$$

$$q_i = \begin{bmatrix} \Delta_{x,i} \\ \Delta_{y,i} \\ L_i \end{bmatrix} = G_i(T_i) \quad (3.2)$$

$L_i$  is evaluated as the mean value of the tendon lengths:

$$L_i = \frac{l_{1,i} + l_{2,i} + l_{3,i}}{3} \quad (3.3)$$

Two methods are developed in order to find the configuration variables: the first one presented uses the standard PCC parameterisation  $\Phi_i$  and  $\theta_i$  with the equations from 3.4 to 3.6.  $d_i$  is the diameter of the section of the arm.

$$\theta_i = 2 \frac{\sqrt{l_{1,i}^2 + l_{2,i}^2 + l_{3,i}^2 - l_{1,i}l_{2,i} - l_{2,i}l_{3,i} - l_{1,i}l_{3,i}}}{d(l_{1,i} + l_{2,i} + l_{3,i})} L_i \quad (3.4)$$

$$\Phi_i = \tan^{-1} \left( \frac{\sqrt{3}(l_{3,i} + l_{2,i} - 2l_{1,i})}{3(l_{2,i} - l_{3,i})} \right) \quad (3.5)$$

$$\begin{aligned} \Delta_{x,i} &= \theta_i d_i \cos(\Phi_i), \\ \Delta_{y,i} &= \theta_i d_i \sin(\Phi_i). \end{aligned} \quad (3.6)$$

The equations of the new modelling are implemented from 3.7 to 3.9 showing that the final result is the same

$$\begin{aligned} \delta_{1,i}(\Phi_i) &= d_i \cos(\Phi_i), \\ \delta_{2,i}(\Phi_i) &= -d_i \cos(\Phi_i), \\ \delta_{3,i}(\Phi_i) &= d_i \sin(\Phi_i), \\ \delta_{4,i}(\Phi_i) &= -d_i \sin(\Phi_i). \end{aligned} \quad (3.7)$$

$$\begin{aligned} L_{1,i}(\Phi_i, \theta_i, \delta_{1,i}L_i) &= L_{0,i} + \delta L_i - \theta_i d_i \cos(\Phi_i), \\ L_{2,i}(\Phi_i, \theta_i, \delta_{2,i}L_i) &= L_{0,i} + \delta L_i + \theta_i d_i \cos(\Phi_i), \\ L_{3,i}(\Phi_i, \theta_i, \delta_{3,i}L_i) &= L_{0,i} + \delta L_i - \theta_i d_i \cos(\Phi_i), \\ L_{4,i}(\Phi_i, \theta_i, \delta_{4,i}L_i) &= L_{0,i} + \delta L_i + \theta_i d_i \cos(\Phi_i). \end{aligned} \quad (3.8)$$

$$\begin{aligned} \Delta_{x,i} &= \frac{L_{2,i} - L_{1,i}}{2} = \theta_i d_i \cos(\Phi_i), \\ \Delta_{y,i} &= \frac{L_{4,i} - L_{3,i}}{2} = \theta_i d_i \sin(\Phi_i). \end{aligned} \quad (3.9)$$

### 3.2.2 From configuration to cartesian

The change of variables of this subsection is illustrated in 3.10. The output of the previous subsection becomes the input in order to find the cartesian coordinates  $X_i = [x_i \ y_i \ z_i]^T$  which delineate the central point of the section designated as the  $i$ -th.

$$q_i = [\Delta_{x,i} \ \Delta_{y,i} \ L_i]^T \Rightarrow X_i = [x_i \ y_i \ z_i]^T \quad (3.10)$$

$$X_i = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = F_i(q_i) \quad (3.11)$$

The variables  $\Delta_{x,i}$  and  $\Delta_{y,i}$  are combined to obtain the general orientation  $\Delta_i$  useful for the next computational passages.

$$\Delta_i = \sqrt{\Delta_{x,i}^2 + \Delta_{y,i}^2} \quad (3.12)$$

The quantity  $R_{i-1}^i$  represents the rotational matrix relating the  $i$ -th section to the preceding one, labelled as the  $(i-1)$ -th section.  $t_{i-1}^i$  contains the cartesian coordinates of the  $i$ -th section with respect of the  $(i-1)$ -th reference system located at the correspondant tip of section and illustrated in 3.1 as  $S_{i-1}$ . The first reference system is the global one and is called  $S_0$  and the others are consequential along the central axis of the robot. Matrix operations permit the evaluation of the triad of axes of the end effector, designated as  $S_{end}$ , in relation to the global reference system. This enables the determination of the orientation. Similarly, the global cartesian coordinates of the end effector are stored in  $t_{end}$ . Consequently, the position is also determined.

$$R_{i-1}^i = \begin{bmatrix} 1 + \frac{\Delta_{x,i}^2}{\Delta_i^2} \left( \cos\left(\frac{\Delta_i}{d_i}\right) - 1 \right) & \frac{\Delta_{x,i}\Delta_{y,i}}{\Delta_i^2} \left( \cos\left(\frac{\Delta_i}{d_i}\right) - 1 \right) & \frac{-\Delta_{x,i}}{\Delta_i} \sin\left(\frac{\Delta_i}{d_i}\right) \\ \frac{\Delta_{x,i}\Delta_{y,i}}{\Delta_i^2} \left( \cos\left(\frac{\Delta_i}{d_i}\right) - 1 \right) & 1 + \frac{\Delta_{y,i}^2}{\Delta_i^2} \left( \cos\left(\frac{\Delta_i}{d_i}\right) - 1 \right) & \frac{-\Delta_{y,i}}{\Delta_i} \sin\left(\frac{\Delta_i}{d_i}\right) \\ \frac{\Delta_{x,i}}{\Delta_i} \sin\left(\frac{\Delta_i}{d_i}\right) & \frac{\Delta_{y,i}}{\Delta_i} \sin\left(\frac{\Delta_i}{d_i}\right) & \cos\left(\frac{\Delta_i}{d_i}\right) \end{bmatrix} \quad (3.13)$$

$$t_{i-1}^i = \frac{d_i(L_{0,i} + \delta L_i)}{\Delta_i^2} \cdot \begin{bmatrix} \Delta_{x,i} \left( 1 - \cos\left(\frac{\Delta_i}{d_i}\right) \right) \\ \Delta_{y,i} \left( 1 - \cos\left(\frac{\Delta_i}{d_i}\right) \right) \\ \Delta_i \sin\left(\frac{\Delta_i}{d_i}\right) \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \quad (3.14)$$

## 3.3 Inverse kinematics

The goal of the IK is to determine the necessary tendon lengths to achieve a desired position and orientation of the end-effector. Typically the IK is more complex, often involving solving nonlinear equations, as multiple configurations can yield the same end-effector position. This complexity increases with the number of d.o.f.

### 3.3.1 From cartesian to configuration

In order to evaluate the configuration space from a given demand expressed in cartesian coordinates, it is fundamental to determine the Jacobian, which expresses the relationship between the rates of change of the configuration variables (velocities) and the rates of change of the end effector. In these equations, only the position is considered, and the focus is on a single section, therefore the Jacobian matrix is 3x3.

$$J_i(q_i) = \frac{\partial F_i(q_i)}{\partial q_i} = \begin{bmatrix} \frac{\partial x_i}{\partial \Delta_{x,i}} & \frac{\partial x_i}{\partial \Delta_{y,i}} & \frac{\partial x_i}{\partial L_i} \\ \frac{\partial y_i}{\partial \Delta_{x,i}} & \frac{\partial y_i}{\partial \Delta_{y,i}} & \frac{\partial y_i}{\partial L_i} \\ \frac{\partial z_i}{\partial \Delta_{x,i}} & \frac{\partial z_i}{\partial \Delta_{y,i}} & \frac{\partial z_i}{\partial L_i} \end{bmatrix} \quad (3.15)$$

The result is achieved through an iterative process utilising a numerical method, such as the jacobian transpose method or the jacobian pseudo-inverse method. For this case the last one has been developed in 3.16 where  $J_i^+$  represent the pseudo-inverse of the jacobian matrix,  $\Delta X_i$  is the distance between the new desired position and the old one and  $\Delta q_i$  is the corresponding variation in the configuration parameters. To solve the IK with a numerical and iterative method, is necessary to solve the FK in order to find the function  $F_i$  shown in equation 3.11.

$$\Delta q_i = J_i^+(q_i) \Delta X_i \quad (3.16)$$

In order to invert the Jacobian matrix, the methodology developed in reference [30] is employed in equation 3.17, thereby facilitating the generation of a stiffness matrix that enables the independent control of changes across each section.  $K_i$  is the diagonal matrix of stiffness that needs to be tuned in order to balance the changes in the robot.

$$J_i^+ = K_i^{-1} J_i^T (J_i K_i^{-1} J_i^T)^{-1} \quad (3.17)$$

### 3.3.2 From configuration to tendon lengths

As with FK, also in this case there are two possible implementations in order to evaluate the tendon lengths starting from configuration space. The first one is the jacobian approach (equations 3.18 and 3.19), while the second one is based on geometric relationships and is illustrated from equation 3.20 to 3.25. In the code, the latter solution is developed because it is computationally faster in the logic of numerical calculations.

In terms of jacobian method, the logic is the same as the previous case but the jacobian matrix is defined with respect of input and output variables of this case.

$$J_{2,i}(T_i) = \frac{\partial G_i(T_i)}{\partial T_i} = \begin{bmatrix} \frac{\partial \Delta_{x,i}}{\partial l_{1,i}} & \frac{\partial \Delta_{x,i}}{\partial l_{2,i}} & \frac{\partial \Delta_{x,i}}{\partial l_{3,i}} \\ \frac{\partial \Delta_{y,i}}{\partial l_{1,i}} & \frac{\partial \Delta_{y,i}}{\partial l_{2,i}} & \frac{\partial \Delta_{y,i}}{\partial l_{3,i}} \\ \frac{\partial L_i}{\partial l_{1,i}} & \frac{\partial L_i}{\partial l_{2,i}} & \frac{\partial L_i}{\partial l_{3,i}} \end{bmatrix} \quad (3.18)$$

$$\Delta T_i = J_{2,i}^+(T_i) \Delta q_i \quad (3.19)$$

For the method implemented without the pseudo-inverse jacobian matrix, the equations of the FK have been implemented, but the inputs have been replaced with the outputs. First, the parameters  $\theta_i$  and  $\Phi_i$  of the PCC model have been implemented:

$$\theta_i = \Delta_i \quad (3.20)$$

$$\begin{cases} \Phi_i = \arccos\left(\frac{\Delta_{x,i}}{\Delta_i}\right), \text{ for } \Delta_{y,i} > 0 \\ \Phi_i = -\arccos\left(\frac{\Delta_{x,i}}{\Delta_i}\right), \text{ for } \Delta_{y,i} < 0 \end{cases} \quad (3.21)$$

In order to avoid burden on the final equations, four additional parameters are evaluated in 3.22 using trigonometric relationships.

$$\begin{aligned} A_i &= \theta_i \frac{d}{2} \cos(\Phi_i), \\ B_i &= \theta_i \frac{d}{2} \sin(\Phi_i), \\ C_i &= 1 - \cos(\theta_i), \\ D_i &= \begin{bmatrix} \cos(\Phi_i) C_i \\ \sin(\Phi_i) C_i \\ \sin(\theta_i) \end{bmatrix} \end{aligned} \quad (3.22)$$

The rotational matrix of the  $i$ -th section with respect to  $(i-1)$ -th reference system, is evaluated using the PCC standard parameters:

$$R_{i-1}^i = \begin{bmatrix} -\cos(\Phi_i^2) C_i + 1 & -\sin(\Phi_i) - \cos(\Phi_i) C_i + 1 & \cos(\Phi_i) \sin(\theta_i) \\ -\sin(\Phi_i) \cos(\Phi_i) C_i + 1 & -\sin(\Phi_i^2) C_i + 1 & \sin(\Phi_i) \sin(\theta_i) \\ -\cos(\Phi_i) \sin(\theta_i) & -\sin(\Phi_i) \sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \quad (3.23)$$

The equations presented in 3.24 are respectively the cartesian coordinates of the tendons of the  $i$ -th section. The expressions are vectors in three dimensions (3xN, where N is the number of points at which the tendon is modelled) comprising the x-, y-, and z-coordinates, respectively, in the first, second, and last rows.

$$\begin{aligned} w_{1,i} &= R_{i-1}^i \cdot \left[ \begin{bmatrix} \frac{d}{2} \cos(\alpha_0) \\ \frac{d}{2} \sin(\alpha_0) \\ 0 \end{bmatrix} + \frac{L_i - A_i \cos(\alpha_0) - B_i \sin(\alpha_0)}{\theta_i} \cdot D_i \right] \\ w_{2,i} &= R_{i-1}^i \cdot \left[ \begin{bmatrix} \frac{d}{2} \cos\left(\frac{2\pi}{3} + \alpha_0\right) \\ \frac{d}{2} \sin\left(\frac{2\pi}{3} + \alpha_0\right) \\ 0 \end{bmatrix} + \frac{L_i - A_i \cos\left(\frac{2\pi}{3} + \alpha_0\right) - B_i \sin\left(\frac{2\pi}{3} + \alpha_0\right)}{\theta_i} \cdot D_i \right] \\ w_{3,i} &= R_{i-1}^i \cdot \left[ \begin{bmatrix} \frac{d}{2} \cos\left(\frac{4\pi}{3} + \alpha_0\right) \\ \frac{d}{2} \sin\left(\frac{4\pi}{3} + \alpha_0\right) \\ 0 \end{bmatrix} + \frac{L_i - A_i \cos\left(\frac{4\pi}{3} + \alpha_0\right) - B_i \sin\left(\frac{4\pi}{3} + \alpha_0\right)}{\theta_i} \cdot D_i \right] \end{aligned} \quad (3.24)$$



To evaluate the total length of each tendon, the sum of the Euclidean distances between successive points  $[x_j \ y_j \ z_j]$  along the curve in three dimensional space is adopted. The corresponding values are contained in  $[w_{1,i(1),j} \ w_{1,i(2),j} \ w_{1,i(3),j}]$  for the first tendon of the  $i$ -th section and so on for the other tendons.  $N$  represents the number of points on the curve, with  $j$  denoting the index of these points.

$$\begin{aligned}
 l_{1,i} &\approx \sum_{j=1}^{N-1} \sqrt{(w_{1,i(1),j+1} - w_{1,i(1),j})^2 + (w_{1,i(2),j+1} - w_{1,i(2),j})^2 + (w_{1,i(3),j+1} - w_{1,i(3),j})^2} \\
 l_{2,i} &\approx \sum_{j=1}^{N-1} \sqrt{(w_{2,i(1),j+1} - w_{2,i(1),j})^2 + (w_{2,i(2),j+1} - w_{2,i(2),j})^2 + (w_{2,i(3),j+1} - w_{2,i(3),j})^2} \\
 l_{3,i} &\approx \sum_{j=1}^{N-1} \sqrt{(w_{3,i(1),j+1} - w_{3,i(1),j})^2 + (w_{3,i(2),j+1} - w_{3,i(2),j})^2 + (w_{3,i(3),j+1} - w_{3,i(3),j})^2}
 \end{aligned} \tag{3.25}$$

### 3.4 Control

The IK implemented for the total robot is showed in figure 3.5(a). In order to control both the position and the orientation of the end effector, it is necessary to utilise six d.o.f. as inputs, as three d.o.f. are insufficient. Three d.o.f. are for the robot-end point while the other three are for the gripper-point that allows to control the orientation since this point is a rigid offset from the previous one in accordance to the orientation of the third and last section of the robot. The configuration space of the entire robot requires nine parameters, three for each section, and the same is for the tendon lengths space. For these reasons, the total jacobian matrix  $J(q)$  evaluated in 3.15 is no more 3x3 but 6x9 because the input for the FK is  $\mathbf{q} = [\Delta x_1, \Delta y_1, L_1, \Delta x_2, \Delta y_2, L_2, \Delta x_3, \Delta y_3, L_3]^T \in \mathbb{R}^9$  and the output  $\mathbf{X} = [x_{robot-end}, y_{robot-end}, z_{robot-end}, x_{gripper}, y_{gripper}, z_{gripper}]^T \in \mathbb{R}^6$ . In analogy the second jacobian matrix  $J_2(T)$  is 9x9 and the tendon lengths space is defined  $\mathbf{T} = [l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8, l_9]^T \in \mathbb{R}^9$ .

The controller adopted to move the robot is the 6 d.o.f. 3d connexion space-mouse in figure 3.5(b) that allows to control separately the translations and the rotations along the three main axes. The output of the joystick is an array  $\mathbf{Joystick} = [t_1, t_2, t_3, r_1, r_2, r_3, B_1, B_2]^T \in \mathbb{R}^8$ . The initial six elements are analogous and correspond to the three translations and the three rotations along the primary axis. The final two signals are digital and correspond to the buttons of the joystick. All commands are incremental, whereby in rest conditions, all signals are set to 0. These commands update the desired position (goal) in Cartesian space, which is then input for the IK. Figure 3.5(d-e) illustrates the behaviour of the robot in response to the two primary input types when operated independently. It is possible to control the end effector with the joystick by combining all six d.o.f. commands.

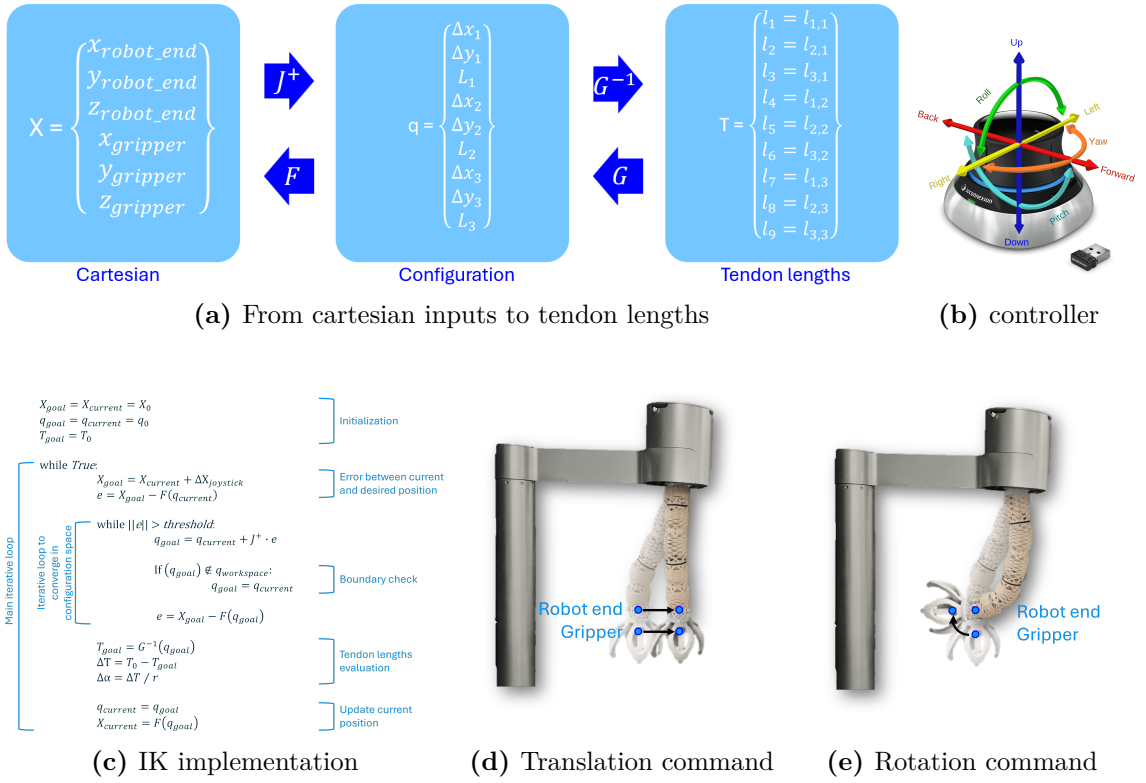


Figure 3.5: Joystick inputs to control the robot.

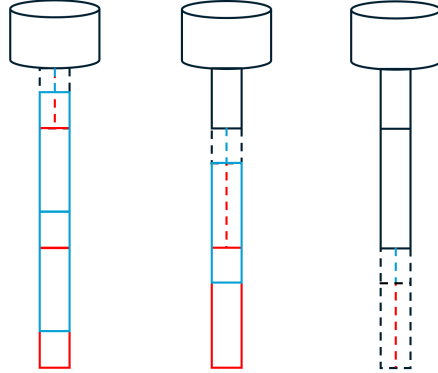
### 3.4.1 Boundaries

The IK developed performs since the goal does not exceed the WS defined by the boundary conditions. Given the PCC nature of the robot, it is not feasible to define the boundaries in cartesian space, as this depends on the current configuration. It is possible to reach certain points with a specific orientation of the end effector, but this is not possible with a different one. For this reason, the boundaries have been evaluated using the variables of the configuration space. In particular, a series of tests have been conducted in order to find a relationship between the length of the central axis  $L_i$  and the global orientation defined as  $\Delta_i = \sqrt{\Delta x_i^2 + \Delta y_i^2}$ . The aim of the tests is to find the suitable range of deformation for each section using the open-loop control implemented. It is anticipated that future developments will see the implementation of closed-loop control utilising IMUs, which will enhance the WS through the implementation of error compensation.

#### Compression tests

The objective of the compression test is to identify the configuration that corresponds to the maximum compression along the central and straight axis. The maximum

extension is defined as the initial configuration where the tendons are in tension but the structure of the robot is not yet deformed. This kind of test has been done for each section and in figure 3.6 are showed the qualitative configurations of the robot when each of the three sections is compressed to the lower limit. The table 3.1 illustrates the numerical values obtained with this test.



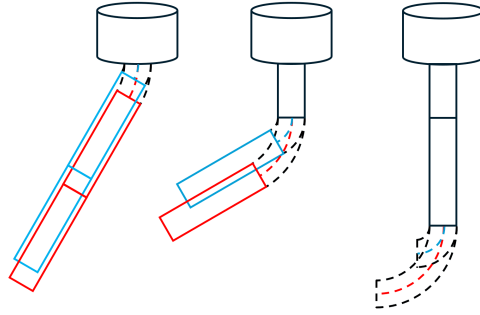
**Figure 3.6:** Evaluation of maximum compression for each section. The blue dotted line represents the lower limit of the central line  $l$  while the red one is the upper limit. In blue and in red are represented the limit configurations of the robot.

Section	$\Delta_i$ [rad]	lower limit $L_i$ [mm]	upper limit $L_i$ [mm]
1	0	55	105
2	0	115	255
3	0	125	240

**Table 3.1:** compression limit conditions for each sections.

### Bending tests

The objective of the bending tests is to ascertain the maximum permissible bending for each section in all directions. It is essential to verify all the four bends along  $x$  and  $y$  axis in both directions because, as the tendons are arranged with an angular displacement of  $120^\circ$ , in some directions is easier to test the maximum bending. In order to optimise the movement of the robot, it is essential to consider the most restrictive condition. Once the maximum bending has been identified, it is essential to find the upper and lower limits of the central axis  $L_i$ , that ensure the maximum bending condition is satisfied. To match the computation of the IK with the real robot, all the tendons have to be always in tension. A simplified sketch is illustrated in figure 3.7 and the results are presented in tables 3.2, 3.3, 3.4 for all the three sections.



**Figure 3.7:** Evaluation of maximum bending for each section. The blue dotted line represents the lower limit of the central line  $l$  while the red one is the upper limit. In blue and in red are represented the limit configurations of the robot.

Direction of bending	$\Delta_1$ [rad]	lower limit $L_1$ [mm]	upper limit $L_1$ [mm]
$\Delta_{x+}$	$\pi/6$	65	90
$\Delta_{x-}$	$\pi/6$	65	85
$\Delta_{y+}$	$\pi/6$	65	85
$\Delta_{y-}$	$\pi/4$	75	85
Limits considered	$\pi/6$	65	85

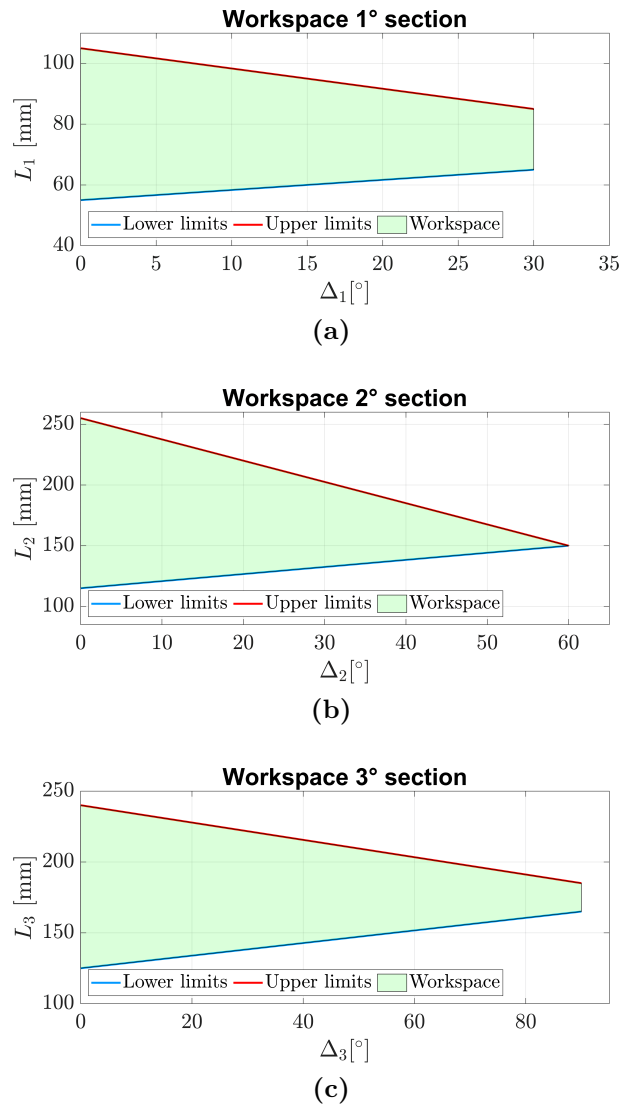
**Table 3.2:** Bending limit conditions for the 1° section.

Direction of bending	$\Delta_2$ [rad]	lower limit $L_2$ [mm]	upper limit $L_2$ [mm]
$\Delta_{x+}$	$\pi/2$	150	190
$\Delta_{x-}$	$\pi/3$	150	150
$\Delta_{y+}$	$\pi/2$	150	170
$\Delta_{y-}$	$\pi/2$	150	170
Limits considered	$\pi/3$	150	150

**Table 3.3:** Bending limit conditions for the 2° section.

Direction of bending	$\Delta_3$ [rad]	lower limit $L_3$ [mm]	upper limit $L_3$ [mm]
$\Delta_{x+}$	$\pi/2$	165	190
$\Delta_{x-}$	$\pi/2$	165	190
$\Delta_{y+}$	$\pi/2$	165	210
$\Delta_{y-}$	$\pi/2$	165	185
Limits considered	$\pi/3$	165	185

**Table 3.4:** Bending limit conditions for the 3° section.

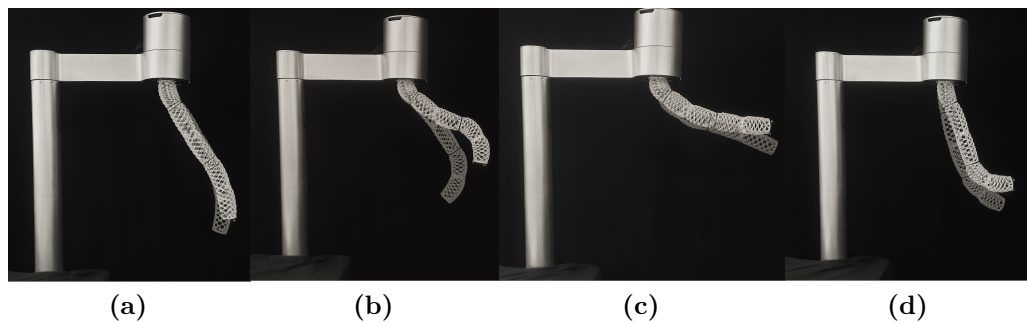


**Figure 3.8:** Workspace in terms of  $L_i$  with respect to bending  $\Delta$ .

The figure 3.8 illustrates the WS for each section evaluated with the tests done. From the condition of  $\Delta_i=0$  to  $\Delta_{i,max}$  there is a linear interpolation in order to identify the proper workspace for each section. This boundary conditions affect the IK code shown in figure 3.5(c). If the request from the user generates a configuration that is not included in the WS, the request is not satisfied and the robot remain fixed in the previous configuration. In terms of  $\Delta_i$  the range increases passing from the 1° to the 3° section because the weight to hold of the robot itself decreases and allows to reach higher bending conditions. This limitation can be partially solved with the implementation of the close-loop control.

To emphasise the significance of the boundary conditions that have been identified,

figure 3.9 illustrates the robot in transparent form, devoid of any boundary conditions with respect to the robot when the boundary conditions are implemented. The discrepancy between the robots in the same figure represents the error between the target and the real configuration that would result in the absence of boundary conditions. In each figure a pose-to-pose command is the input. This behaviour is due to the fact that, without boundaries, the numerical evaluation of the IK could not match the geometry of the real robot. In most of the cases illustrated, the numerical calculations pretend some tendons to be longer than the maximum possible value.

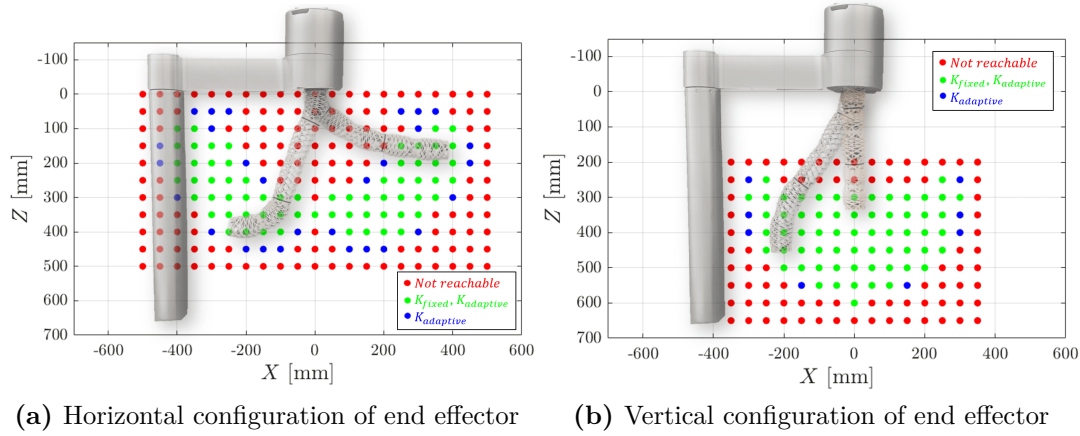


**Figure 3.9:** Comparison between pose to pose control with and without boundary conditions. In transparency the configuration without boundaries with the same goal as the other configuration.

### 3.4.2 Adaptive stiffness matrix

The pseudo-inverse jacobian matrix is evaluated in equation 3.17. The  $K$  diagonal matrix enables the equilibrium of alterations to each parameter within the configuration space. It is not possible to make progress when working with a fixed matrix if certain numerical saturation points are reached. This indicates that the desired position cannot be reached since the while loop employed to minimise the norm (e), as illustrated in figure 3.5(c), is in a state of stalling, resulting in the numbers remaining constant throughout each iteration. In order to go further this limitation, the  $K$  stiffness matrix is adaptive, which means that the values of the diagonal changes in accordance of the current configuration. If a saturation is reached in the  $i$ -th section, this one is made stiffer in order to solve the convergence of the IK adapting the other sections. The advantages of this solution are illustrated in figure 3.10 where the blue points are the one that can be reached using the adaptive stiffness matrix but cannot be reached using the fixed version. The green points show all the points that can be reached in both situations. Another important advantage is the number of iterations that decreases significantly using the adaptive solution. Two conditions are represented, in (a) there is the WS of the robot maintaining the end effector in horizontal position while in (b) the end effector is vertical. Although the figure

represents a plane, it is necessary to conceive the WS in three-dimensional space. The points must be rotated along the central axis of the structure in accordance with the  $x=0$  value indicated on the graphs.



**Figure 3.10:** Grid of points to evaluate the workspace and the the difference using adaptive stiffness matrix.

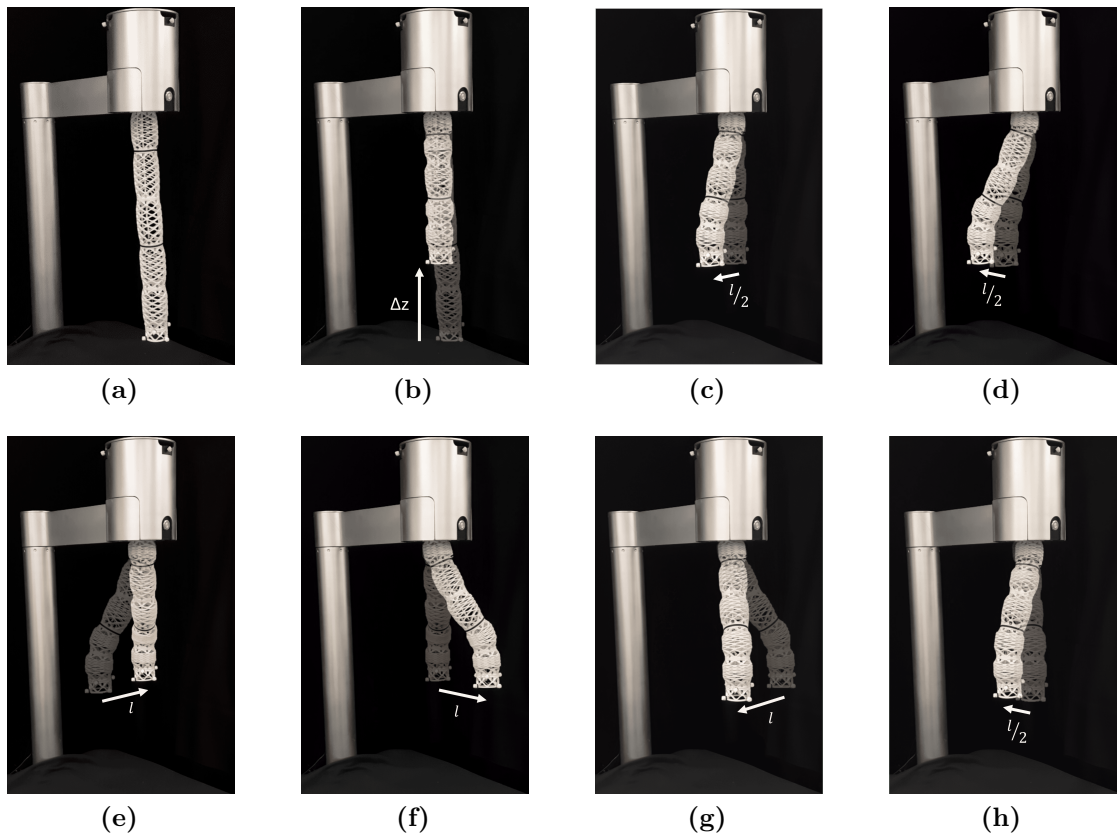
### 3.5 Validation

In order to validate the proposed inverse kinematics model for the soft manipulator, a series of experimental tests have been conducted with the objective of evaluating the accuracy and precision of the control system. In view of the intrinsic flexibility and continuous deformation of soft manipulators, it is important to ensure that the model accurately predicts the position of the robotic arm in order to facilitate effective manipulation tasks. The validation process entailed a comparison between the target positions, derived from the IK solution, and the current ones. The objective of the experiments is to quantify the discrepancy between the intended and actual positions, thereby providing insight into the efficacy of the open-loop control strategy employed to regulate the motion of the manipulator. Two distinct tests have been devised for the purpose of assessing the accuracy of the manipulator in different conditions. The discrepancy between the intended and actual trajectories is quantified by recording the real-time positions of the manipulator using an optical tracking system. It consist on six motion capture cameras (Optic Prime 13) that can recognise white and lighting spheres called motion capture markers, they are arranged radially along the three main sections of the robot and especially at the tip of the soft arm. Three markers are also applied to the support structure in order to give the ground reference system for all the others coordinates recorded. The markers on each segment have been defined as a rigid body in the 'Motive' Software (Optical motion capture

software from OptiTrack). The errors obtained from these tests are the principal indicator of the validity of the IK model developed and the effectiveness of the open-loop control strategy.

### 3.5.1 Square trajectory

The manipulator is required to follow trajectories that are square in shape, with the squares located on three distinct planes within the three-dimensional WS. Two sizes of squares have been tested for each plane. Following the desired trajectory, the end effector is required to maintain the vertical position. The objective of these tests is to evaluate the performance of the model in executing complex motions that require precise control in multiple directions. This test has been devised with the objective of evaluating translation behaviour. The sequence of the required movements of the robot are showed in figure 3.11. The first three frames (a-c) illustrates the movements that allows to reach the starting point where the square trajectory starts to be recorded from the optical tracking system.



**Figure 3.11:** Square trajectory represented in frames.



The table 3.5 illustrates the amplitude of the movements for each trajectory.  $\Delta_z$  is the initial vertical displacement to reach the height of each plane while the parameter  $l$  is the side of the square. Since the initial vertical condition of the robot in figure 3.11(a) corresponds to 600 mm of z-coordinate in the reference system of the arm, the three planes are located respectively at 350, 320 and 400 mm in this reference system. For each of these layers, squares of 100 and 200 mm size are tested.

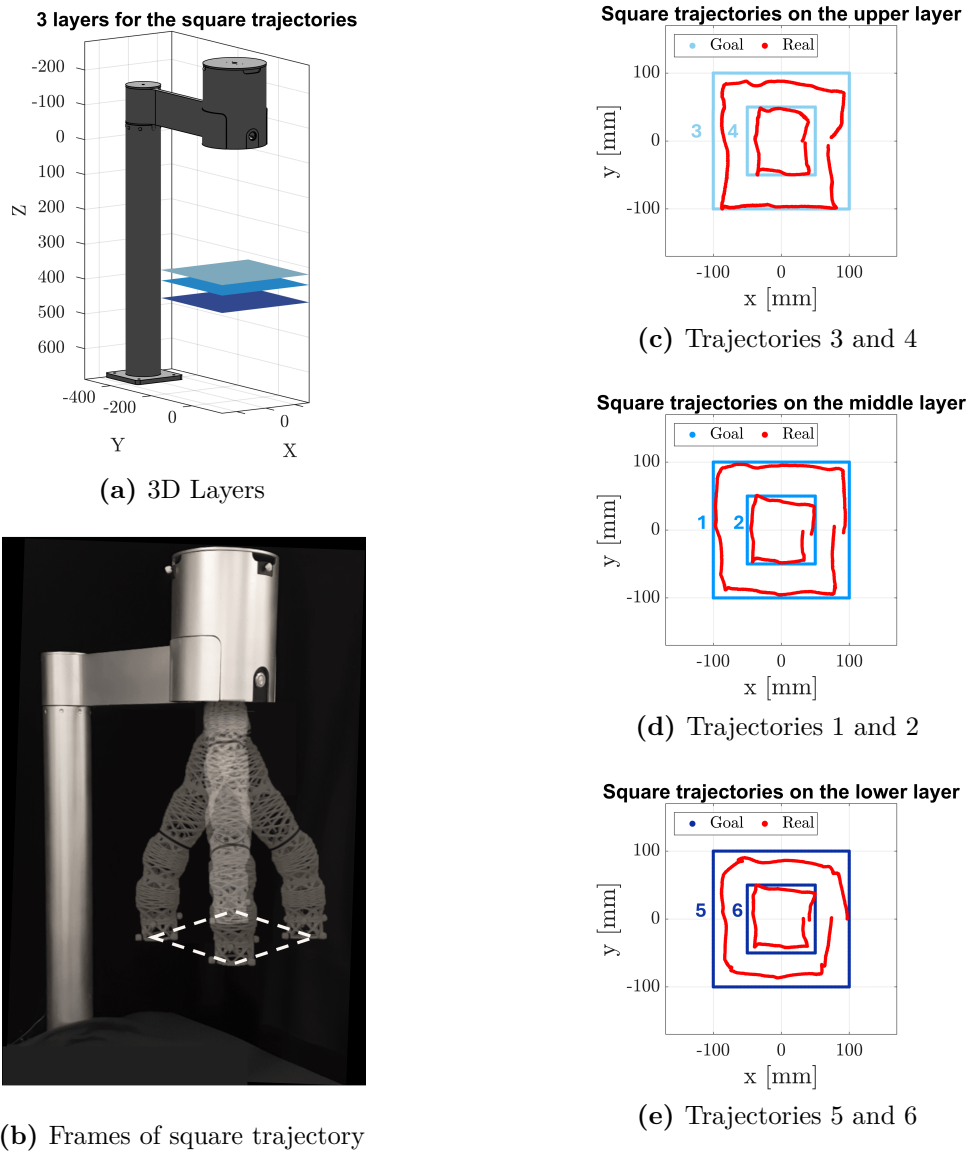
Trajectory	$\Delta_z$ [mm]	$l$ [mm]
1	250	200
2	250	100
3	280	200
4	280	100
5	200	200
6	200	100

**Table 3.5:** Parameterisation for square trajectories.

The figure 3.12 illustrates the square trajectories studied in terms of WS explored and trajectory comparison between the goal and the real behaviour. Subfigure (a) illustrates a 3D representation of the manipulator, highlighting three different layers where the square trajectories perform. These layers, located at different heights along the Z axis, represent the WS regions analysed for trajectory accuracy. Subfigure (b) displays the soft manipulator following one of the square trajectories, showing the configuration of the system during the test. Subfigures (c-e) compare the desired (goal) and actual (real) trajectories on each plane. In each subfigure, the comparison between the goal and real trajectories highlights the deviations observed in the open-loop control performance. The analysis across different planes and trajectory sizes provides insight into the effectiveness of the control strategy in maintaining the desired path, providing a clear perspective on the positional accuracy of the manipulator across different workspace regions.

From the graphs presented, the general behaviour along the middle layer, i.e. trajectories 1 and 2, is better compared to the other cases. This result is probably due to the fact that this height is in the middle of the WS where the robot performs better. On the upper layer, for example, the configuration of the robot is close to its fully compressed configuration which means close to the boundaries. On the other hand, for the trajectories 5 and 6 performed on the lower layer, the weight of the arm is no located in a small central area (e). The accuracy of open-loop control is adversely affected by an increase in the dimensions of the square, due to a shift of the centre of gravity away from the central axis with coordinates  $x=0$  and  $y=0$ .

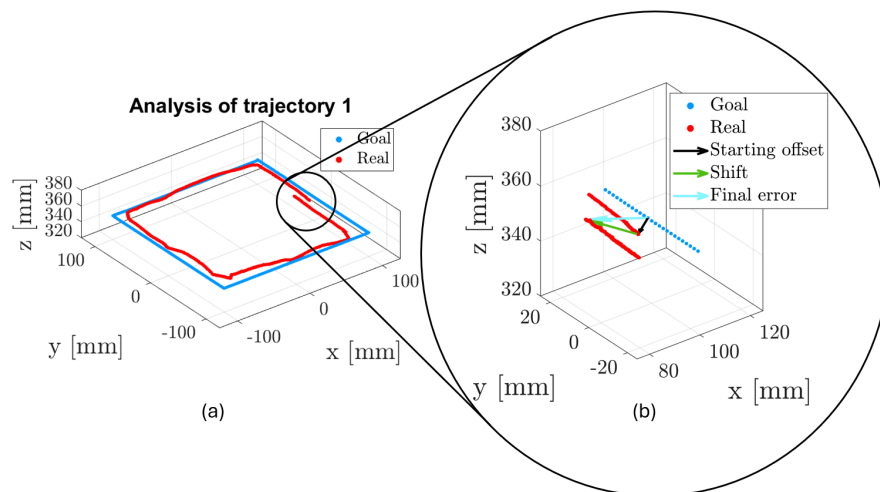
In order to make a quantitative analysis of this trajectories some parameters and distances are taken into account and shown in figures 3.13- 3.14.



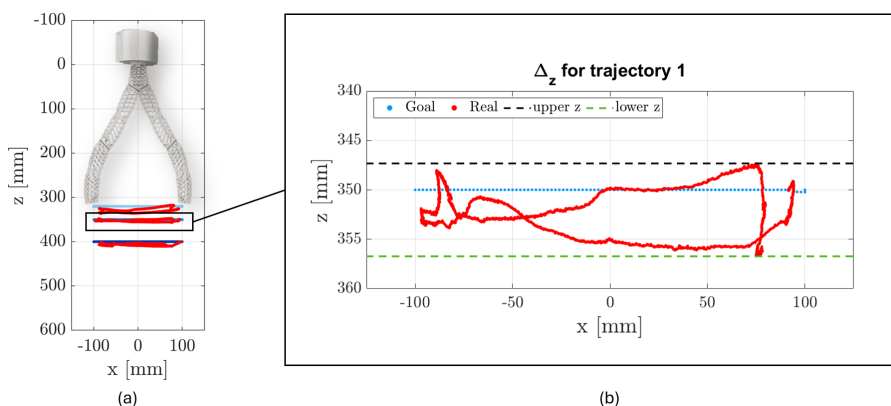
**Figure 3.12:** Layers used for square trajectories and analysis for every plane. Comparison between goal and real trajectory. In (a) there is the illustration of the three different planes considered in order to show the workspace analysed. (b) shows the real configurations of the robot during the test and the square trajectory is highlighted. (c-e) represent the comparison between the goal and real trajectories in the x-y planes studied.

To make an accurate analysis of the test, the trajectory 1 is taken as reference. Figure 3.13 illustrates the comparison between the desired and the real trajectory with a 3D point of view. Figure 3.14 shows the x-z plane in order to analyse the vertical displacement, which must be null. The parameters considered are:

- **Starting offset:** it represents the distance in 3D space between goal and real position before the square trajectory begins. The black vector is represented in figure 3.13(b). This configuration is showed in figure 3.11(c) and corresponds to the offset in initial positioning. This parameters allows to evaluate the accuracy to reach a goal in the space with the end effector in vertical orientation following straight lines and not a pose-to-pose command.
- **Shift:** it represents the distance in 3D space between the real initial position and the final one after the trajectory has been done. The green vector is represented in figure 3.13(b). This configuration is showed in figure 3.11(h) and corresponds to the shift from the initial positioning and the final position reached after the trajectory test. This parameter allows to evaluate the repeatability of the system because measure the shift in the space of two points with the same target but reached in two different times and with different commands.
- **Final error:** it represents the distance in 3D space between goal and real position when the test finishes. The cyan vector is represented in figure 3.13(b). This configuration is showed in figure 3.11(h) and corresponds to the offset in final position. This parameter enables the overall accuracy of the trajectory to be evaluated.
- $\Delta_z$ : it represents the vertical distance between the planes "upper z" and "lower z" displayed in figure 3.14(b). These two planes are respectively aligned to the minimum and maximum z-value recorded by the optical tracking system. Given that all six trajectories are constrained to move in a plane, it can be assumed that this parameter should be equal to zero.



**Figure 3.13:** Square analysis of trajectory 1. In (b) the zoom allows to highlight the parameters taken into account in order to evaluate the performance.



**Figure 3.14:** Square analysis of trajectory 1, offset in  $z$  direction.

The figures illustrate just the case of trajectory 1 in order to define the parameters but the numerical values for all the six cases are showed in table 3.6.

The results of the tests are contained in table 3.6. With regard to the "starting offset" distance, the values for trajectories 3 and 4 are elevated in comparison to the others. This discrepancy can be attributed to the substantial compressibility demanded by the tests conducted on the upper layer. The deformable structure of the arm can affect the positioning behaviour. Looking at the figure 3.12(c) is possible to see that the error between the goal and the effective pose decreases along the trajectory. For the remaining tests, the error is close to 1 cm.

In consideration of the "Shift" vector, the results are quite in accordance with the previous parameter but with an inverse correlation. The maximum value is 24.1 mm in case of trajectory 5 that corresponds to the minimum value of the starting offset. This indicates that the "final error" after the square trajectory is affected by this two variables. Similarly to the previous case, the value of 24.1 mm of trajectory 5 is elevated in comparison to the others and, a part from this, the global shift is in within 10 mm and in general with lower values. This analysis indicates that the repeatability of Shift vector is superior compared to the accuracy of the Starting offset parameter. The exact values are affected by the configuration evaluated by the IK solution and this is why there is no a robust correlation between all the tests. The value of  $\Delta_z$  is influenced by the weight of the soft arm. In larger squares, the weight is distributed less uniformly in the central area, resulting in an increased error when maintaining a constant height. This behaviour is to be expected, given the absence of feedback from sensors and the lack of implementation of any form of weight compensation. For this parameter the middle plane is the ideal one. Final error and  $\Delta_z$  are correlated to the size of the squares, with the values for the larger squares being consistently more elevated across all the planes.

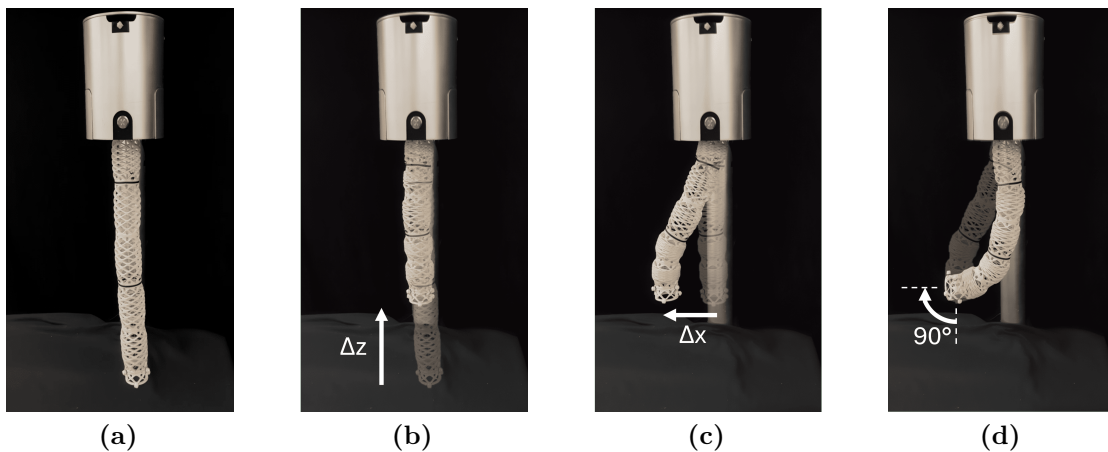
Trajectory	Starting offset [mm]	Shift [mm]	Final error [mm]	$\Delta_z$ [mm]
1	10.5	13.0	23.4	9.4
2	10.0	8.5	18.4	6.2
3	28.1	6.2	34.3	19.0
4	19.4	4.4	14.9	8.7
5	3.6	24.1	27.7	10.4
6	9.3	8.3	17.5	7.0

**Table 3.6:** Analysis for square trajectories.

In conclusion the order of magnitude of the error recorded with this tests is around 1/2 cm. It is essential to proceed with the subsequent tests in order to gain a deeper understanding of the impact of the rotation command on the validation process. It is only after the results of these tests have been analysed that any definitive conclusions can be drawn regarding the validation tests.

### 3.5.2 90° bending trajectory

The objective of this test is to evaluate the performance in executing complex motions that require precise control in multiple directions. Specifically, the manipulator is commanded to perform 90-degree bends starting from various positions in 3D space. This test has been devised with the objective of evaluating rotational movements maintaining the same position of the tip of the robot. Combining the results of this tests with the previous ones, it is possible to have a general idea of the intrinsic error of the system in different configurations. The figure 3.15 shows the frames of the test, (a-c) represent the positioning movement in the 3D space before the bending manoeuvre, which is displayed in subfigure (d).



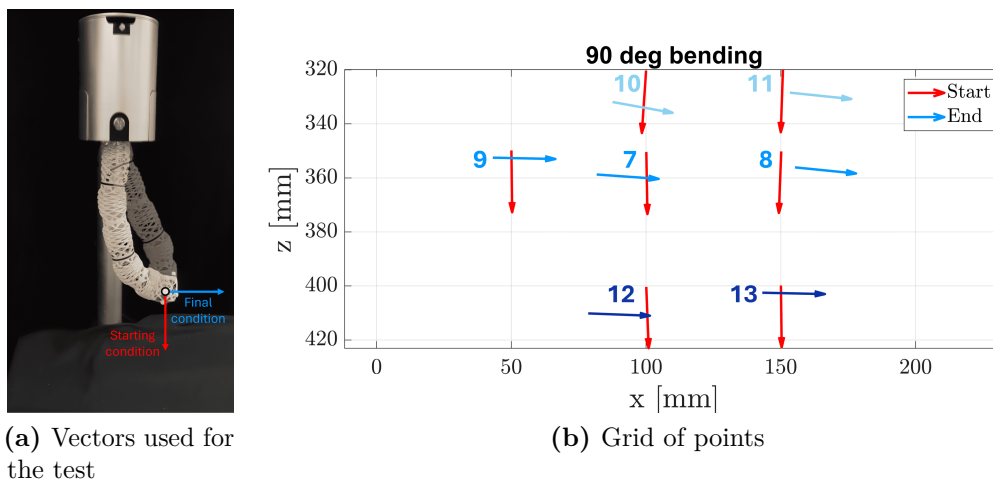
**Figure 3.15:** 90° bending trajectory represented in frames.

The test entails reaching seven distinct points in the x-z plane while maintaining the vertical orientation of the end effector. Subsequently, a 90° bend is to be performed, with the cartesian position remaining unaltered. The table 3.7 shows the parameterisation of the seven trajectories tested.

In order to record the orientation of the end effector with the optical tracking system, three markers are applied to the final section of the robot, in a manner analogous to that employed in the preceding tests to create a rigid body. At the extremity of the arm, a rigid component with the same shape as the structure is affixed in order to construct an additional section with the same orientation as the last one. Three more markers are arranged radially to this extra section, in this way is possible to evaluate the cartesian coordinates and evaluate the orientation combining the data of the two sections considered. The setup is recognisable in figure 3.15.

Trajectory	$\Delta_z$ [mm]	$\Delta_x$ [mm]
7	250	100
8	250	150
9	250	50
10	280	100
11	280	150
12	200	100
13	200	150

**Table 3.7:** parameterisation of the starting point for 90° bending trajectories.

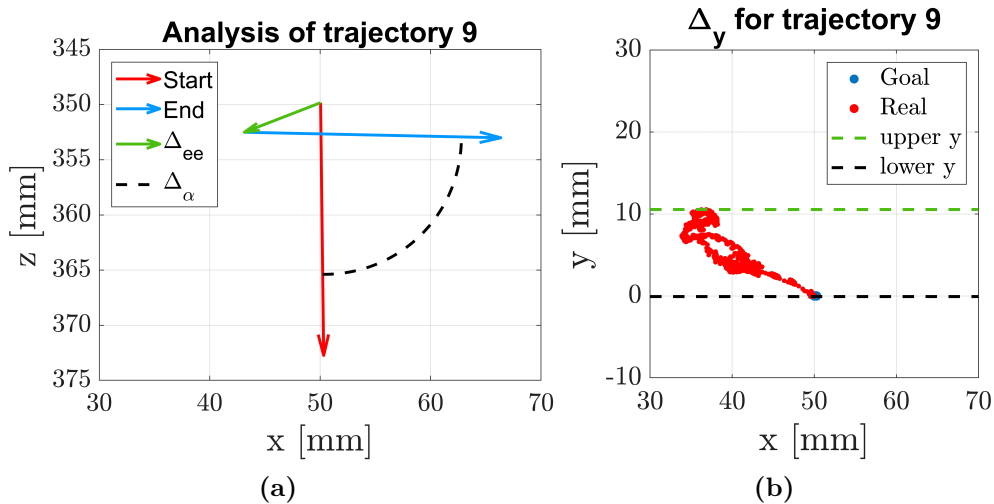


**Figure 3.16:** Analysis of 90° bending trajectory. In figure (a) there is the front point of view of trajectory 7. The red arrow represents the direction of the end effector at the start of the bending test whereas the blue one is the final orientation. In figure (b) there is the grid of points tested.

The figure 3.16 illustrates the two vectors that characterise the orientation of the extremity of the soft arm. In contrast with the preceding case, only two vectors are represented, rather than all of the data recorded, in order to ensure the readability of the graphs. The red vectors represent the initial position while the blue ones represent the final position. The three blue vectors used in subfigure (b) are consistent with the three planes at different heights also employed in the preceding tests. (a) shows the real configuration of the robot for trajectory 7, in (b) there is the grid of points tested.

The parameters taken into account in order to find a quantitative estimation of the error are:

- $\Delta_{ee}$ : it is a vector that connect the position of the end effector from the initial position to the final one. It should be zero since the input commands of the IK is a pure rotation around this point.
- **Starting offset**: it represents the initial angular position that is the angular error since the initial target is  $0^\circ$ . It is the orientation of the red vector shown in figure 3.17(a).
- $\Delta_\alpha$ : it is the real measured angular displacement done during the test with respect to the desired  $90^\circ$ .
- $\Delta_y$ : as for the previous tests, this parameter evaluates the movement along y-axis that should be null.



**Figure 3.17:** Analysis of  $90^\circ$  bending trajectory 9. In figure (a) there are the parameters involved in order to evaluate the performance. In figure (b) there is the top point of view to visualize the movement along the y-axis of the end effector.

The results of all the trajectories tested are presented in table 3.8.  $\Delta_{ee}$  is significantly elevated for trajectories 7, 10 and 12. All the other values are consistently lower within 10 mm. The angular starting offset is comparable between all the cases subjected to analysis without a general trend. The total angular displacement recorded during the test is the parameter  $\Delta_\alpha$ , and the overall angle measured with initial displacement of  $\Delta_x=150$  mm perform better than the trajectories characterised by  $\Delta_x=100$  mm in all the planes considered. The tendency of the movement along the y-axis represented by  $\Delta_y$  is in accordance to this behaviour. In general, tests that have in common the displacement  $\Delta_x=100$  mm perform worse, for this reason maybe the cause of the error is the configuration of the robot that is reached before the bending test with this value of translation along the x-direction.

Trajectory	$\Delta_{ee}$ [mm]	Starting offset [°]	$\Delta_\alpha$ [°]	$\Delta_y$ [mm]
7	24.2	1.0	84.9	22.2
8	8.6	2.2	86.5	17.2
9	8.3	0.7	88.1	10.6
10	17.1	3.9	83.8	10.2
11	9.6	2.0	86.0	12.4
12	24.9	2.3	85.6	18.4
13	8.3	0.7	88.1	10.6

**Table 3.8:** Analysis for 90° bending trajectories.

The optical tracking system has a resolution that is significantly higher than the distances evaluated. However, there is a possibility of a minor discrepancy in the recorded data due to the methodology employed. In order to record the trajectory of the end effector, three markers have been attached radially to the final section of the robot. The tracking system used the markers to construct a rigid body and evaluate the coordinates at its centre to find the coordinates of the central point of the last section.

The accuracy obtained is not comparable to the ones of traditional and rigid robot. The aim of the soft arm developed is not being as precise as the rigid robots but it is to perform safely collaborative tasks with human interactions. In addition, the design of the gripper allows to grasp objects without a very high precision. When it is combined with the human-in-the-loop control utilising the joystick, it becomes possible to undertake the tasks described in the following section to validate the capabilities of the system.

In conclusion the accuracy needs to be improved with closed-loop control which is an ongoing area of research but the PCC model and the IK developed can be considered validated with these limitations.





# Chapter 4

## Results and applications

This chapter presents the experimental validation and practical applications of the developed robotic system. The objective of the results is to demonstrate the efficacy of the soft robotic gripper and the corresponding control strategies in authentic real-world tasks. This chapter is structured around two key experiments. The first is a pick and place task involving two cubes, and the second is a feeding task designed to highlight the capabilities of the soft robot compared to a rigid one.

This chapter is divided into several key sections, which are outlined below. Initially, a series of tests was conducted with the objective of evaluating the user-friendliness of the system, with a particular emphasis on its ease of use for individuals lacking prior experience. Subsequently, a comparative analysis was conducted between the soft arm and a traditional rigid arm. The distinctive capabilities of the soft manipulator, particularly its capacity for collaborative and delicate operations, have been demonstrated through its utilisation in a highly invasive task that has been performed with both arms.

The final section presents a number of applications of the developed robot and gripper, demonstrating the versatility and capabilities of the soft manipulator for a range of complex and collaborative tasks. This chapter represents the conclusion of the work presented in this thesis, demonstrating the potential of soft robotics in advancing human-robot interaction and enhancing task performance in challenging environments.

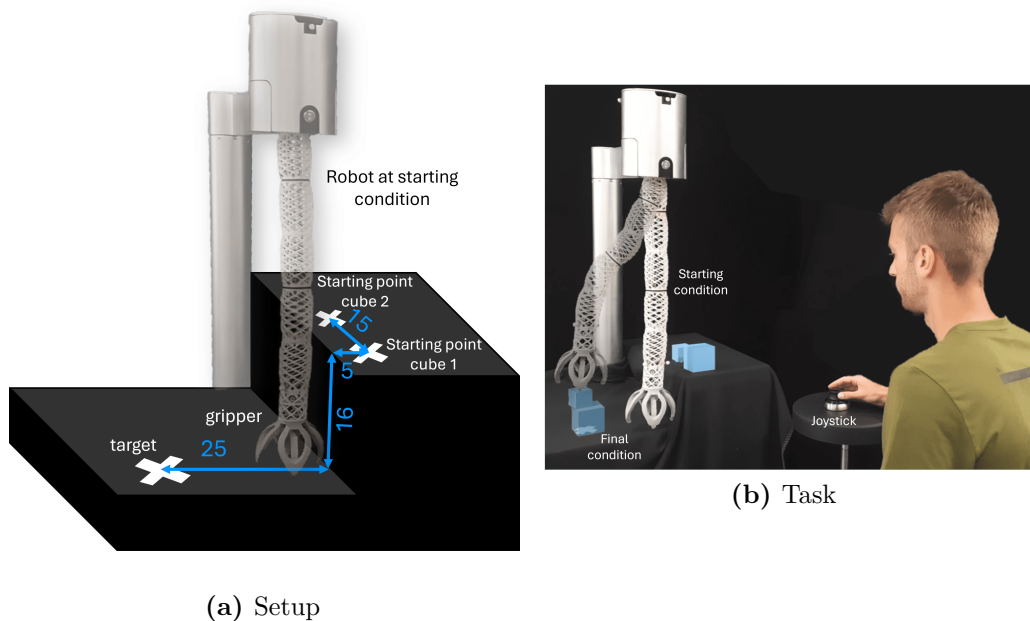
### 4.1 Pick and place of two cubes

The objective of this experiment is to evaluate the user-friendliness of the developed control system for individuals with no prior experience of using it. The study involves five different users, each of whom completed five attempts. The time taken to complete each attempt has been recorded, and a standard questionnaire has been provided to validate the user interaction.

### 4.1.1 Experimental setup and task

The experimental setup is showed in figure 4.1(a). The distances are expressed in centimeters. The aim of the test is to perform a pick and place operation with two 3D printed PLA cubes, with a specified level of precision. The cubes, differing in size, with one measuring 50 mm and the other 75 mm, are initially positioned in close proximity to one another on two designated crosses marked on the table. The robot begins in a fully extended vertical position. The user is required to first grasp the larger cube and place it on a target cross, followed by the positioning of the smaller cube on top of the larger one. The illustration of the task is presented in figure 4.1(b), the user is positioned close to the arm and the robot is showed both in initial and final configuration to highlight the task.

The task is neither collaborative nor particularly challenging for the robot, but it allows to evaluate both the human interaction with the system and the precision of pick and place task. This is due to the fact that the target is clearly defined with a cross, and the cubes must be positioned one above the other. The simplicity of the task makes it suitable for evaluating how intuitively users can control the robot using the joystick interface, especially those with little to no experience in robotics.



**Figure 4.1:** Experimental setup of the pick and place of 2 cubes task. (a) The 2 crosses on the higher plane represent the starting position of the cubes whereas the cross on the lower plane is the final target where the cubes must be stacked. The distances are in cm. (b) The task with starting and final position presented in transparency.

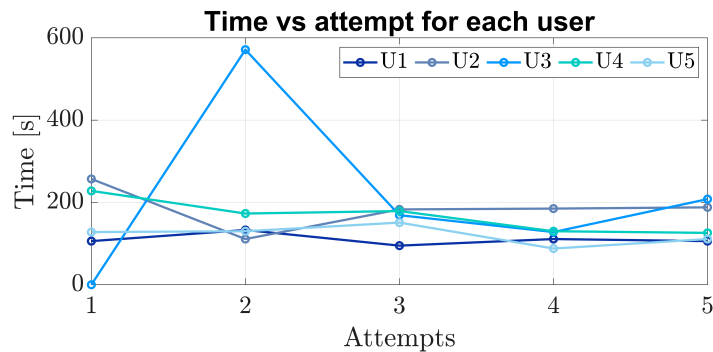
In addition to its function as a test of fundamental functionality, the task permits a more profound investigation of the responsiveness to input precision of the system, task completion time, and user comfort in operating the robot.

#### 4.1.2 Data analysis and conclusions

Table 4.1 presents the time required for each user to complete the task. The first attempt of User 3 has been deleted from the analysis because of a mechanical failure of the robot. These data are showed graphically in figure 4.2. The results demonstrate that the time required to complete the task reaches a point of saturation. This implies that after an initial learning period, the time taken by users to perform the task stabilises and does not significantly decrease with further attempts. This behaviour indicates that the joystick control system is highly intuitive and user-friendly. This result is particularly important for systems intended for human-robot interaction, as it suggests that even users with no prior experience can quickly become able to control the robot, making it suitable for a wide range of operators, including those without technical backgrounds. The completion time for the task exhibits a range of 2 to 3 minutes, contingent upon the time spent approaching and grasping the cube but it depends on random external factors like the manner to approach the cubes.

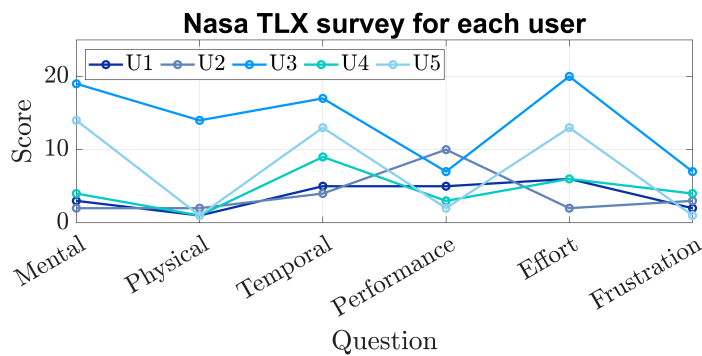
Attempt	User 1	User 2	User 3	User 4	User 5
1	1.46	4.17	Fail	3.48	2.08
2	2.13	1.51	9.31	2.53	2.10
3	1.35	2.43	2.49	2.59	2.31
4	1.51	2.45	2.08	2.10	1.28
5	1.46	3.08	3.28	2.06	1.51

**Table 4.1:** Time in [mm.ss] required to do the task for each user.

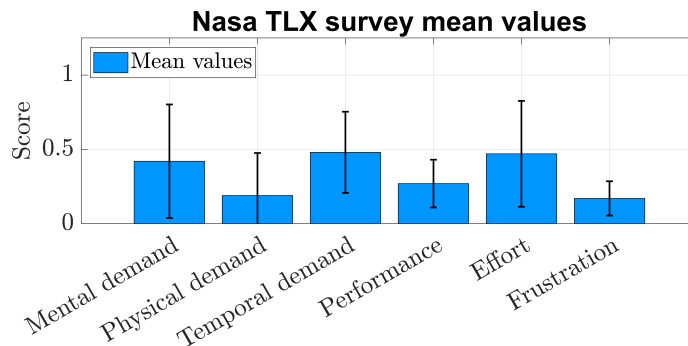


**Figure 4.2:** Time vs attempts for the cubes task in a plot to evaluate the time and the ease of use of the robot using the joystick as teleoperation system.

After few trials, users are able to perform the task at a consistent pace with minimal cognitive load or effort. This is a clear indication of an effective control design, as the users are able to teleoperate the system with minimal effort and efficiently translate their intended movements into robot actions. In order to measure the real cognitive load and effort the Nasa TLX, a standard questionnaire, has been submitted to the users. It is a widely used subjective tool for multidimensional assessments of perceived workload. The tool is adopted to evaluate the demands of a task, measuring the effectiveness of a task, system. It consist on six questions with a scale of marks from 1 to 20. A higher score indicates a greater workload. The sample is presented in the AppendixB.1 in figure B.1 [31].



(a) Analysis for each user



(b) Analysis of average values

**Figure 4.3:** Results of the Nasa TLX survey. In (a) there is the trend of the score for each user while in (b) there is the average of the scores of all users for each question, with the standard deviation displayed.

The figure 4.3 illustrates the data collected in table 4.2. The ratings of the user 3 are extremely distant from those of other users. Combining this result with the time spent for the test, this particular user results the one that has been more uncomfortable with the soft manipulator. Looking at the mean values for each

question, the first three questions demonstrate that the control is not physically demanding due to the smooth joystick control. However, it is evident that a certain degree of concentration and time is required to complete the task but always above the 50% of the entire scale.

The worst recorded value is 5.4 for the performance. This is likely attributable to the precision of the open loop control in comparison to the expectation of the users that all input commands would be perfectly aligned with the behaviour of the robot. The effort required to complete the task is indicative of a moderate level, while the frustration experienced is minimal.

The standard deviation is pretty high and it is likely due to the fact that the test has been done by just five different users. Frustration and performance ratings are the most consistent, with relatively low standard deviations while mental demand and effort show the most variability.

Question	Mental demand	Physical demand	Temporal demand	Performance	Effort	Frustration
User 1	3.0	1.0	5.0	5.0	6.0	2.0
User 2	2.0	2.0	4.0	10.0	2.0	3.0
User 3	19.0	14.0	17.0	7.0	20.0	7.0
User 4	4.0	1.0	9.0	3.0	6.0	4.0
User 5	14.0	1.0	13.0	2.0	13.0	1.0
Average	8.4 42.0%	3.8 19.0%	9.6 48.0%	5.4 27.0%	9.4 47.0%	3.4 17%
$\sigma$ - StD	38.2%	28.6%	27.3%	16.1%	35.6%	11.5%

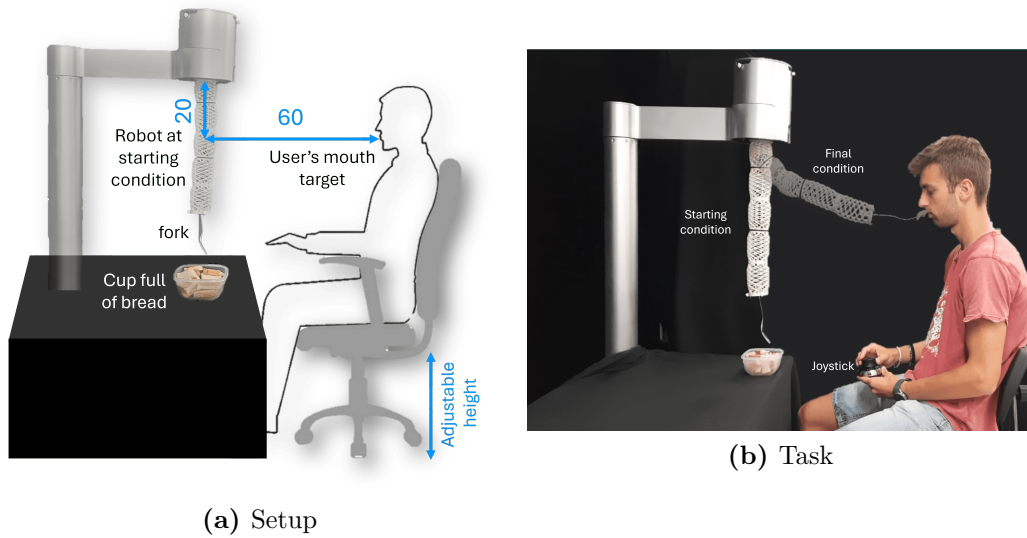
**Table 4.2:** Analysis of the survey for the pick and place of 2 cubes task.

## 4.2 Feeding task in order to compare the soft robot with a rigid robot

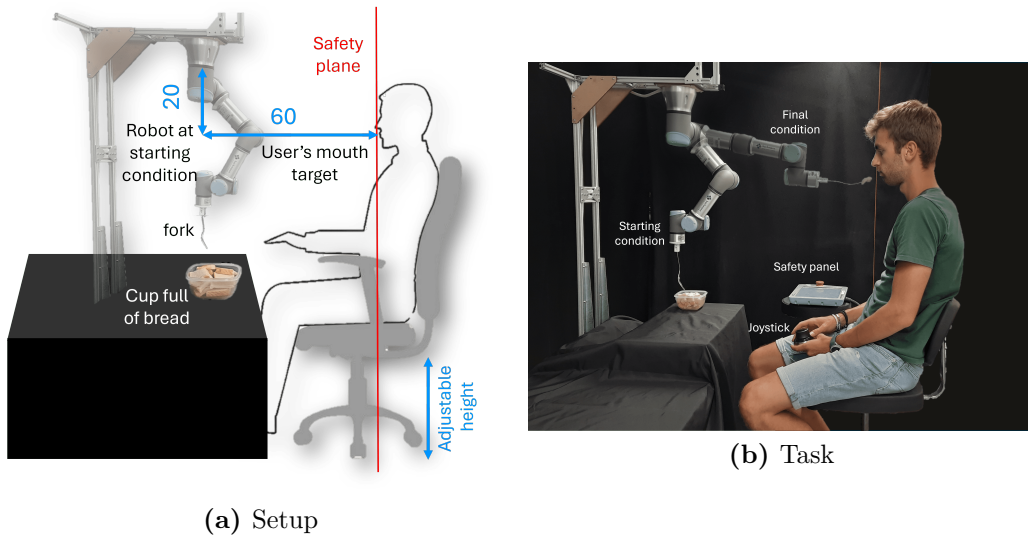
The objective of this experiment is to evaluate the user experience of performing a highly invasive task. In particular, the feeding task has been considered and performed with both the Helix soft manipulator and UR3 rigid robot in order to facilitate a comparison. The study involves five participants, each of whom completed three trials. Subsequently, a questionnaire has been provided to validate the test.

### 4.2.1 Experimental setup and task

The experimental setup used for the Helix is showed in figure 4.4(a). The objective of the test is to perform a feeding task, specifically grasping pieces of bread from a cup located on the table.



**Figure 4.4:** Experimental setup of feeding task with Helix robot. (a) Setup with target and initial position. (b) shows the user ready to do the test. The task is illustrated thanks to the starting and final position, the last one in transparency.



**Figure 4.5:** Experimental setup of feeding task with UR3 robot. (a) Setup with target and initial position. (b) shows the user ready to do the test. The task is illustrated thanks to the starting and final position, the last one in transparency.

In this iteration of the experiment, the gripper is replaced by a fork. In order to ensure consistency for all users, the chair is adjustable in height, allowing the mouth of the user, which is the target for the task, to be at the same distance with

respect to the arm. This ensures that all users are able to perform the task in a consistent manner. The robot commences in a calibrated position, defined by a partially compressed configuration. The user is required to take a piece of bread with the fork and then utilise the joystick to operate the soft arm in order to feed themselves. The objective is to maintain the head in a fixed position in order to complete the task by moving the robot and not the head of the person.

Figure 4.5(a) illustrates the setup adopted and the distances from the arm base are the same as the previous case. This analogous setup has been constructed for the Universal Robot 3 (UR3) with the objective of creating a task that is as similar as possible to the previous one in terms of displacement required to finish the task. This allows for a more accurate comparison to be made, as the setup is almost identical. Also in this case, the joystick control has been implemented to teleoperate the rigid manipulator. For safety reason, since the robot is rigid and can hurt the user, the plane that correspond to the maximum workspace of the arm has been identified. The safety plane is a straight red thread with a weight hanging from it represented in figure 4.5(b)

From a control perspective, the task is not particularly challenging as the motion is potentially constrained to a single plane. However, the objective of this test is to assess the confidence of the users when performing invasive tasks.

### 4.2.2 Data analysis and conclusions

The time required to complete the task is no more interesting for this type of test but another questionnaire has been submitted to the users in order to measure the confidence when performing invasive task. A total of seven users participated in the test, three of whom (Users 1, 2 and 5) employed both robots for comparative purposes, while the remaining four (Users 3A, 3B, 4A and 4B) have completed the task with only one robot. In this way, it is possible to make different analysis for the users that have made the comparison and the ones that have done the test with just one manipulator.

In order to measure the feelings of the users with quantitative method, a questionnaire called "Trust between people and automation" has been submitted to all of them, the sample is in the AppendixB.2 in figure B.2 [32]. It consists on twelve questions with a scale of marks from 1 to 7. The initial five questions are designed to prefer low scores, thereby underscoring the confidence in the system. For instance, one question is "I am wary of the system." In contrast, the final seven questions employ a contrasting approach, prompting respondents to indicate their level of agreement with statements such as, "The system provides security.". In this case high scores are in accordance with confidence in the system. In order to ensure consistency, the results have been scaled in a range of -1 to 1, where 1 represents the highest value, zero represents a neutral position, and -1 represents the lowest value. This methodology allows for the evaluation of two distinct scales of percentage: the first comprising positive values,



which indicate favourable outcomes, and the second comprising negative values. The twelve questions have been divided into five correlated latent human factors which group some items together and provide a more comprehensive explanation of behaviour than would be possible by studying each item individually. The five groups are:

- TRUST: It measure if the robor is trustworthy. Questions 1, 4, 11.
- SAFETY: If the user feels safe working next to the robot and confident that the robot will not hurt him. Questions 2, 5, 7.
- FLUENCY: Measurement of the robot contribution to the fluency of interaction. Questions 6, 12.
- COMPETENCE: It is the capability of the user to handling the robot to perform the task. Questions 9, 10.
- EFFORTLESSNESS: It represents the easiness for the user to make the robot do what he wants to accomplish. Questions 3, 8.

Question	1	2	3	4	5	6	7	8	9	10	11	12
User 1 <sub>Helix</sub>	3.0	1.0	2.0	1.0	1.0	6.0	5.0	5.0	6.0	6.0	7.0	7.0
User 1 <sub>UR3</sub>	5.0	5.0	4.0	6.0	6.0	5.0	3.0	4.0	3.0	3.0	4.0	5.0
User 2 <sub>Helix</sub>	4.0	2.0	1.0	2.0	3.0	2.0	3.0	5.0	3.0	3.0	5.0	5.0
User 2 <sub>UR3</sub>	5.0	6.0	4.0	6.0	6.0	2.0	2.0	3.0	3.0	3.0	2.0	2.0
User 3A <sub>Helix</sub>	4.0	5.0	4.0	4.0	3.0	2.0	2.0	5.0	4.0	5.0	4.0	5.0
User 3B <sub>UR3</sub>	1.0	1.0	1.0	4.0	3.0	5.0	3.0	2.0	5.0	5.0	5.0	6.0
User 4A <sub>Helix</sub>	4.0	7.0	2.0	1.0	1.0	6.0	6.0	6.0	6.0	6.0	7.0	6.0
User 4B <sub>UR3</sub>	4.0	3.0	2.0	6.0	3.0	4.0	2.0	5.0	5.0	5.0	3.0	2.0
User 5 <sub>Helix</sub>	2.0	4.0	3.0	6.0	1.0	4.0	4.0	5.0	5.0	6.0	6.0	6.0
User 5 <sub>UR3</sub>	2.0	7.0	6.0	6.0	6.0	3.0	2.0	4.0	6.0	3.0	2.0	2.0
<i>AVG</i> <sub>Helix</sub>	3.4	3.8	2.4	2.8	1.8	4.0	4.0	5.2	4.8	5.2	5.8	5.8
<i>AVG</i> <sub>UR3</sub>	3.4	4.4	3.4	5.6	4.8	3.8	2.4	3.6	4.4	3.8	3.2	3.4

**Table 4.3:** Analysis of the survey for the feeding task.

A preliminary examination of the comparative data for the Helix and UR3 scores across all users reveals an intriguing consistency in the test results for those who have completed both evaluations. The observed outcomes align closely with the overall mean values, suggesting a high degree of correlation between the two tests. In the remaining cases, however, this is not the case. The scores awarded to the first questions by users 3A and 3B are completely at odds with one another. In particular, user 3B appears to exhibit a greater degree of comfort utilising the UR3 than the user

3A using the Helix. Although the rigid robot may appear to demonstrate superior confidence based on these values, the average values obtained from all cases do not support this hypothesis. It is possible that user 3B displays greater confidence with robotic systems than user 3A.

The analysis in terms of average values and standard deviation is represented in two different tables 4.4-4.5, one for each robot. The percentage values with negative sign represent the cases where the mean value is above the neutral condition so they represent negative feedback from the users.

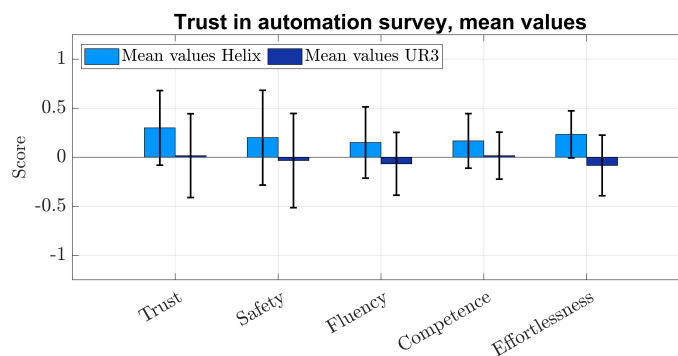
Helix	Trust	Safety	Fluency	Competence	Effortlessness
Average	+30.0%	+20.0%	+15.0%	+16.7%	+23.3%
$\sigma$ - StD	38.0%	48.4%	36.4%	27.8%	24.0%

**Table 4.4:** Analysis of the survey for the feeding task using Helix robot.

UR3	Trust	Safety	Fluency	Competence	Effortlessness
Average	+1.7%	-3.3%	-6.7%	+1.7%	-8.3%
$\sigma$ - StD	42.3%	48.1%	32.0%	24.0%	30.9%

**Table 4.5:** Analysis of the survey for the feeding task using UR3 robot.

Graphically the mean values of the data collected from the experience of the users are illustrated in figure 4.6. It is evident that the results of the Helix are consistently within the optimal range with respect to the UR3 ones for all the metrics taken into account. These good results justify the development of the soft robotics because, for a task that requires an invasive interaction, the soft arm results safer and more suitable instead of a traditional collaborative robot.



**Figure 4.6:** Results of the Trust in automation survey. The bar represents the average of the scores of all users for each metric. The black sign represents the standard deviation.

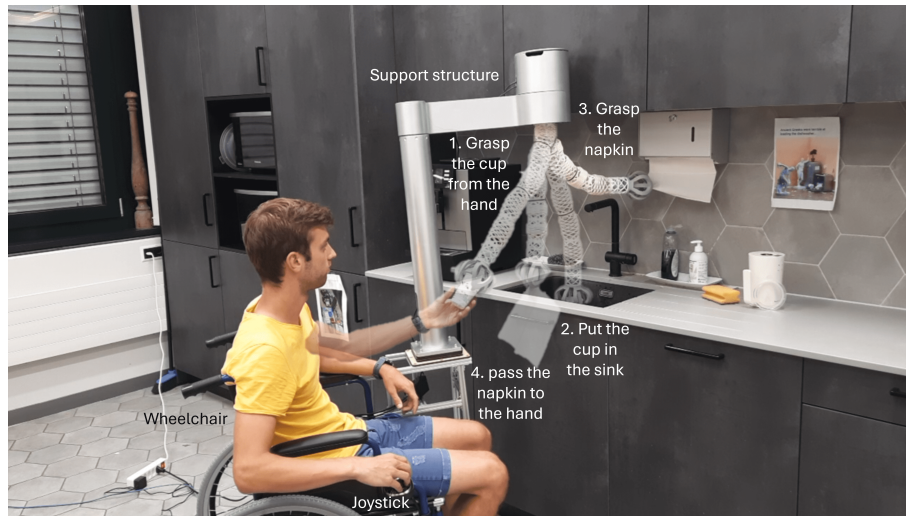
The greatest discrepancy is observed in terms of safety and trust, which is in accordance with the expectations that were formulated at the outset of this study. The data indicate that the Helix is more effortless than the UR3, which is a surprising result. Another important aspect is that the values of the Helix robot are always in the positive range while for the rigid robot it is negative for three cases and positive with small values for the others. The standard deviation is a statistical measure that quantifies the dispersion of values of a variable around its mean. In this context, the mean is the score for each question. The high values observed in the survey are a consequence of the limited number of data points used to calculate the average, which was based on just five numbers. To reduce this value, it would be beneficial to increase the number of users involved in the survey.

A comparison of the use of Helix robot and UR3 robot for performing a highly invasive feeding task revealed a clear preference for the soft arm among the users. The soft robot has been selected primarily due to its ability to provide a greater sense of safety and comfort during the task. This preference can be attributed to the intrinsic design of the soft arm, which is better suited for tasks that involve close and direct interaction with humans. In contrast, rigid manipulators are not typically designed for such collaborative applications, as their mechanical structure lacks the flexibility required for high levels of human interaction. Moreover, rigid robots may occasionally display abrupt and unanticipated movements as they undergo reconfiguration to reach a novel objective, as a consequence of the inverse kinematics calculations. Such sudden alterations in motion can prove disconcerting or even intimidating for users, particularly when they are situated in the same workspace. This contrast in user experience serves to illustrate the limitations of rigid manipulators in collaborative tasks, thereby reinforcing the advantages of soft robotics in scenarios that require sensitive and close human-robot interaction.

### **4.3 Applications and capabilities**

This concluding section brings together the various elements of the thesis, including the joystick control based on the IK and the gripper design, in order to illustrate the capabilities of the Helix and to present a number of potential scenarios in which it can be deployed. The potential of the soft robotic arm as an assistive tool for individuals with disabilities, particularly those who rely on wheelchairs, is demonstrated through the presentation of real-world applications shown in figure 4.7.

Two different scenarios are presented and allows the user to grasp objects too far for the person with limited mobility on the wheelchair, such as the napkin in figure 4.7(a) or the apple from the fridge in (b). The safety and lightweight design of the manipulator make it an appropriate choice for mounting on a wheelchair, allowing for the performance of collaborative tasks with close human interaction.



(a) Grasp the cup and pass the napkin



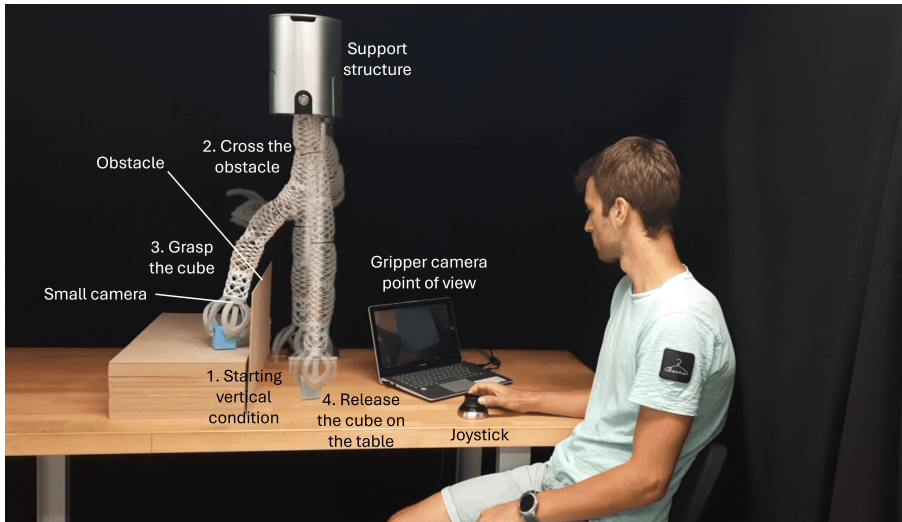
(b) Grasp the apple from the fridge

**Figure 4.7:** Kitchen scenarios highlighting assistance for disabled people using wheelchair. The robot is mounted directly on the wheelchair, there are some configurations of the robot to show the task.

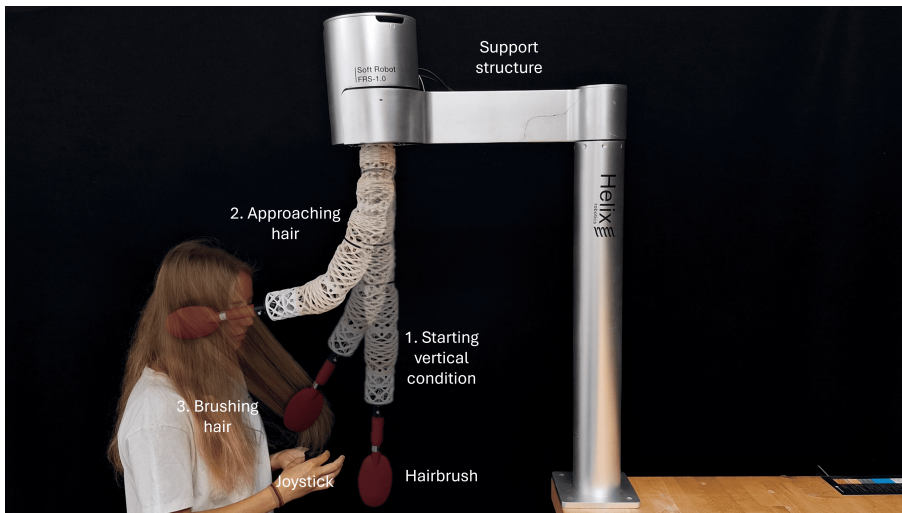
The Helix manipulator, operated via a joystick interface, has been developed with the objective of facilitating a range of activities that promote user independence and enhance the quality of life. One key feature is the integration of a local camera mounted on the gripper, allowing the robot to grasp objects that are out of the point of view of the user, addressing a common limitation in assistive technology. Figure 4.8 shows this capability.

Additionally, the ability to perform everyday tasks such as brushing is showed in

figure 4.9, highlights its adaptability and ease of use. The user selected for this test had no previous experience of robotic systems, but she managed to complete the test feeling comfortable, even with such an invasive task. This scenario perfectly represents the potential application in everyday life for a user that has no experience with robotics in general.



**Figure 4.8:** Use of a local and small camera installed in the gripper in order to reach a point not visible from the user point of view showing the deformation capabilities of the arm.



**Figure 4.9:** Teleoperated brushing task with a close interaction between the user and the soft arm. The hairbrush is mounted instead of the gripper at the tip of the robot.

These demonstrations underscore the practical value of soft robotics in assistive contexts, where safety, flexibility, and intuitive control are paramount. By enabling individuals with limited mobility to perform personal tasks autonomously, this soft robotic arm represents a significant advancement in assistive technologies.

In conclusion, this thesis successfully introduces a novel soft continuum manipulator capable of performing dynamic pick-and-place tasks. The creation of a rapid and adaptable gripper, coupled with a comprehensive examination of the kinematics of the Helix soft manipulator and human interaction control strategies, illustrates the feasibility of the system in real-world scenarios. The integration of these contributions demonstrates the user-friendly control and repeatable performance, which underscores its potential for dynamic tasks and interaction with human operators. This work represents a significant step forward in the field of soft robotics and paves the way for future advancements in adaptive and efficient manipulation systems.



# Appendix A

## Python scripts

Here are the Python code developed and used in this thesis.

### A.1 Main IK code

**Listing A.1:** Main IK code

```
1 # IK FROM CARTESIAN COORDINATES AND ORIENTATION OF THE END EFFECTOR TO
  TENDON LENGTHS
2 # Input: goal to reach using the joystick [x_ee, y_ee, z_ee, x_grip,
  y_grip, z_grip]
3 # Output: differential tendond lengths which is the input for the
  raspberrypi L = [11, 12, 13, 14, 15, 16, 17, 18, 19]
4
5 import numpy as np
6 import keyboard
7 import time
8 import roslibpy
9 from functions_ik import *
10
11 # Connection to the Pi's rosbridge server
12 client = roslibpy.Ros(host='192.168.1.2', port=9090) # The host is the
  Pi's IP address
13 client.run()
14
15 # Calibration
16 # Switch to current control
17 srv = roslibpy.Service(client, '/tendon_transmission_node/
  switch_to_current_control', 'std_srvs/Trigger')
18 req = roslibpy.ServiceRequest()
19 # Add tension and straighten manually the arm
20 srv = roslibpy.Service(client, '/tendon_transmission_node/set_current',
  'helix_transmission_interfaces/SetCurrent')
21 req = roslibpy.ServiceRequest({"current" : -70.0})
```



```

22 # Save calibration file
23 srv = roslibpy.Service(client , '/tendon_transmission_node/
    set_motor_offsets', 'std_srvs/Trigger')
24 req = roslibpy.ServiceRequest()
25 # Switch back to position control
26 srv = roslibpy.Service(client , '/tendon_transmission_node/
    switch_to_position_control', 'std_srvs/Trigger')
27 req = roslibpy.ServiceRequest()
28
29 # Subscribe to the tendon state information topic in order to
    communicate with ROS
30 tendon_listener = roslibpy.Topic(client , '/tendon_transmission_node/
    tendon_states', 'sensor_msgs/msg/JointState')
31 joystick_listener = roslibpy.Topic(client , '/spacenv/joy', '
    sensor_msgs/msg/Joy')
32 tendon_talker = roslibpy.Topic(client , '/tendon_transmission_node/
    commands', 'std_msgs/msg/Float64MultiArray')
33 gripper_talker = roslibpy.Topic(client , '/helix_gripper_node/
    command_increment', 'std_msgs/msg/Float64')
34
35 # Function to communicate with the motors
36 def tendon_callback(message):
37     global joint_names
38     joint_names = message['name']
39     global tendon_positions
40     tendon_positions = message['position']
41
42 # Function to communicate with the 6 d.o.f. joystick
43 def joystick_callback(message):
44     global axes
45     axes = message['axes']
46     global buttons
47     buttons = message['buttons']
48
49 tendon_listener.subscribe(tendon_callback)
50 joystick_listener.subscribe(joystick_callback)
51
52 # Function to rotate along X axis in order to align the reference
    system of the real robot
53 def rotx(theta):
54     theta = np.radians(theta)
55     return np.array([
56         [1, 0, 0],
57         [0, np.cos(theta), -np.sin(theta)],
58         [0, np.sin(theta), np.cos(theta)]
59     ])
60
61 # Function to evaluate the cartesian coordinates of the griper using
    the coordinates of the ee and the 3 rotational vectors
62 def update_gripper_position(P, Q, rotation_increment):

```

```

63     Q_x, Q_y, Q_z = Q
64     delta_theta_x_degrees, delta_theta_y_degrees, delta_theta_z_degrees
        = rotation_increment
65     delta_theta_x = np.deg2rad(delta_theta_x_degrees)/2
66     delta_theta_y = np.deg2rad(delta_theta_y_degrees)/2
67     delta_theta_z = np.deg2rad(delta_theta_z_degrees)/2
68
69     Rx_increment = np.array([[1, 0, 0],
70                               [0, np.cos(delta_theta_x), -np.sin(
71     delta_theta_x)],
72                               [0, np.sin(delta_theta_x), np.cos(
73     delta_theta_x)]]))
74     Ry_increment = np.array([[np.cos(delta_theta_y), 0, np.sin(
75     delta_theta_y)],
76                               [0, 1, 0],
77                               [-np.sin(delta_theta_y), 0, np.cos(
78     delta_theta_y)]]))
79     Rz_increment = np.array([[np.cos(delta_theta_z), -np.sin(
80     delta_theta_z), 0],
81                               [np.sin(delta_theta_z), np.cos(
82     delta_theta_z), 0],
83                               [0, 0, 1]])
84
85     R_increment = np.dot(Rz_increment, np.dot(Ry_increment,
86     Rx_increment))
87     direction = np.array([Q_x - P[0], Q_y - P[1], Q_z - P[2]])
88     rotated_direction = np.dot(R_increment, direction)
89     updated_Q = P + rotated_direction
90     return updated_Q
91
92     print("Wait a moment before using the joystick.")
93     # Tuning parameters to change the speed of the robot and the threshold
94     # accepted in the IK
95     gain_trasl = .0015
96     gain_rot = 1
97     threshold = 0.001 #[m]
98
99     # Define the initial position of the end effector and the gripper
100    end_effector_x = 0 #[m]
101    end_effector_y = 0 #[m]
102    end_effector_z = .600 #[m]
103    L_gripper = .050 #[m]
104    gripper_x = 0 #[m]
105    gripper_y = 0 #[m]
106    gripper_z = end_effector_z + L_gripper #[m]
107    end_effector = np.array([end_effector_x, end_effector_y, end_effector_z
108    ])
109    gripper = np.array([gripper_x, gripper_y, gripper_z])
110    old_joystick = np.array([end_effector_x, end_effector_y, end_effector_z
111    , gripper_x, gripper_y, gripper_z])

```

```

102 new_joystick = np.array([end_effector_x, end_effector_y, end_effector_z
103     , gripper_x, gripper_y, gripper_z])
104 # Initial vertical condition in configuration state and tendon lengths
105 DxDyL_new = [0.0001, 0.0001, .105, 0.0001, 0.0001, .255, 0.0001,
106     0.0001, .240]
107 DxDyL_old = [0.0001, 0.0001, .105, 0.0001, 0.0001, .255, 0.0001,
108     0.0001, .240]
109 Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3 = DxDyL_new
110 L_new = [0.105, 0.105, 0.105, 0.255, 0.255, 0.255, 0.240, 0.240, 0.240]
111 L0 = [0.105, 0.105, 0.105, 0.255, 0.255, 0.255, 0.240, 0.240, 0.240]
112 l1, l2, l3, l4, l5, l6, l7, l8, l9 = L_new
113 # Upper and lower limits to the variables
114 D1_max, D2_max, D3_max = np.pi/6, np.pi/3, np.pi/2
115 limits = np.array([D1_max, D2_max, D3_max])
116 # Matrix of stiffness in order to balance the changes along the
117     sections
118 K_elements = [.1 * L_new[6] / L_new[0], .1 * L_new[6] / L_new[0], 1 *
119     L_new[6] / L_new[0], .1 * L_new[6] / L_new[3], .1 * L_new[6] /
120     L_new[3], 1 * L_new[6] / L_new[3], .1, .1, 1]
121 K = np.diag(K_elements)
122 K_inv = np.linalg.inv(K)
123 # Evaluation of the endeffector cartesian coordinates and the Jacobian
124     matrix in symbolic way to speed up the loop of IK
125 t3_c_result, J_result = endeffector_Jacobian_function(L_gripper)
126 print("\nNow you can use the joystick.")
127 print("type 'r' to return in vertical position and 'e' to end the loop
128     .")
129 exit_flag = False
130 while not exit_flag :
131     # End the IK loop
132     if keyboard.is_pressed('e') :
133         exit_flag = True
134     # Come back in calibrated position
135     if keyboard.is_pressed('r') :
136         end_effector_x = 0 #[m]
137         end_effector_y = 0 #[m]
138         end_effector_z = .600 #[m]
139         L_gripper = .050 #[m]
140         gripper_x = 0 #[m]
141         gripper_y = 0 #[m]
142         gripper_z = end_effector_z + L_gripper #[m]
143         end_effector = np.array([end_effector_x, end_effector_y,
144     end_effector_z])

```

```

143     gripper = np.array([gripper_x, gripper_y, gripper_z])
144     old_joystick = np.array([end_effector_x, end_effector_y,
end_effector_z, gripper_x, gripper_y, gripper_z])
145     new_joystick = np.array([end_effector_x, end_effector_y,
end_effector_z, gripper_x, gripper_y, gripper_z])
146
147     DxDyL_new = [0.0001, 0.0001, .105, 0.0001, 0.0001, .255,
0.0001, 0.0001, .240]
148     DxDyL_old = [0.0001, 0.0001, .105, 0.0001, 0.0001, .255,
0.0001, 0.0001, .240]
149     Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3 = DxDyL_new
150     L_new = [0.105, 0.105, 0.105, 0.255, 0.255, 0.255, 0.240,
0.240, 0.240]
151     L0 = [0.105, 0.105, 0.105, 0.255, 0.255, 0.255, 0.240, 0.240,
0.240]
152     l1, l2, l3, l4, l5, l6, l7, l8, l9 = L_new
153     tendon_talker.publish(roslibpy.Message({'data':
[0,0,0,0,0,0,0,0,0]}))
154
155     # Open the gripper typing a button on the joystick
156     if buttons[0] == 1:
157         gripper_talker.publish(roslibpy.Message({'data': 0.005}))
158         print("button to open the gripper pushed")
159         time.sleep(0.001)
160
161     # Close the gripper typing a button on the joystick
162     if buttons[1] == 1:
163         gripper_talker.publish(roslibpy.Message({'data': -0.005}))
164         print("button to close the gripper pushed")
165         time.sleep(0.001)
166
167     translation = np.dot(rotx(180), [axis * gain_trasl for axis in axes
[:3]])
168     end_effector = end_effector + translation
169     gripper = gripper + translation
170     rotation = np.dot(rotx(180), [axis * gain_rot for axis in axes
[3:]])
171     gripper = update_gripper_position(end_effector, gripper, rotation)
172     new_joystick = np.concatenate((end_effector, gripper))
173
174     # Solve the IK if the goal is changed
175     if np.any(new_joystick != old_joystick):
176         delta_joystick = new_joystick - old_joystick
177         DxDyL_new, new_joystick = cartesian_configuration(new_joystick,
old_joystick, J_result, K_inv, DxDyL_new, DxDyL_old, limits,
t3_c_result, threshold)
178         L_new = configuration_tendon(DxDyL_new)
179         DeltaL = L_new - L0
180         print("End effector position [x, y,z, x_grip, y_grip, z_grip
]:", new_joystick)

```

```

181     old_joystick = new_joystick.copy()
182     DxDyL_old = DxDyL_new.copy()
183     [end_effector_x, end_effector_y, end_effector_z, gripper_x,
184     gripper_y, gripper_z] = old_joystick
185     end_effector = old_joystick[:3]
186     gripper = old_joystick[3:]
187
187     j=DeltaL.tolist()
188     tendon_talker.publish(roslibpy.Message({'data': j}))
189
190 print("\nFinal End effector position [x, y,z, x_grip, y_grip, z_grip
191       ]:", new_joystick)
192 print("Final configuration [Dx1 Dy1 L1 Dx2 Dy2 L2 Dx3 Dy3 L3]:",
193       DxDyL_new)
194 print("Final tendon lengths [l1 l2 l3 l4 l5 l6 l7 l8 l9]:", L_new)

```

## A.2 Functions

**Listing A.2:** functions-ik code imported in the main one

```

1 # Useful functions to import in order to solve the IK of the robot
2 import numpy as np
3 from sympy import symbols, sqrt, cos, sin, Matrix, lambdify
4 from numpy.linalg import norm
5
6 # Function that evaluate the endeffector cartesian coordinates and the
7 # Jacobian matrix in symbolic way
8 def endeffector_Jacobian_function(L_gripper):
9     D1 = sqrt(Dx1**2 + Dy1**2)
10    R1 = Matrix([[1 + (Dx1**2 / D1**2) * (cos(D1) - 1),
11                (Dx1 * Dy1 / D1**2) * (cos(D1) - 1),
12                (-Dx1 / D1) * sin(D1)],
13               [(Dx1 * Dy1 / D1**2) * (cos(D1) - 1),
14                1 + (Dy1**2 / D1**2) * (cos(D1) - 1),
15                (-Dy1 / D1) * sin(D1)],
16               [(Dx1 / D1) * sin(D1),
17                (Dy1 / D1) * sin(D1),
18                cos(D1)]])
19    t1 = Matrix([(L1 / D1**2) * Dx1 * (cos(D1) - 1),
20                (L1 / D1**2) * Dy1 * (cos(D1) - 1),
21                (L1 / D1**2) * D1 * sin(D1)])
22
23    D2 = sqrt(Dx2**2 + Dy2**2)
24    R12 = Matrix ([[1 + (Dx2**2 / D2**2) * (cos(D2) - 1),
25                  (Dx2 * Dy2 / D2**2) * (cos(D2) - 1),
26                  (-Dx2 / D2) * sin(D2)],
27                 [(Dx2 * Dy2 / D2**2) * (cos(D2) - 1),
28                  1 + (Dy2**2 / D2**2) * (cos(D2) - 1),

```

```

28         (-Dy2 / D2) * sin(D2) ],
29         [(Dx2 / D2) * sin(D2),
30         (Dy2 / D2) * sin(D2),
31         cos(D2) ]])
32 t12 = Matrix([(L2 / D2**2) * Dx2 * (cos(D2) - 1),
33              (L2 / D2**2) * Dy2 * (cos(D2) - 1),
34              (L2 / D2**2) * D2 * sin(D2)])
35 t2 = t1 + R1 * t12
36 R2 = R1 * R12
37
38 D3 = sqrt(Dx3**2 + Dy3**2)
39 R23 = np.array([(1 + (Dx3**2 / D3**2) * (cos(D3) - 1),
40                (Dx3 * Dy3 / D3**2) * (cos(D3) - 1),
41                (-Dx3 / D3) * sin(D3)],
42                [(Dx3 * Dy3 / D3**2) * (cos(D3) - 1),
43                 1 + (Dy3**2 / D3**2) * (cos(D3) - 1),
44                 (-Dy3 / D3) * sin(D3)],
45                [(Dx3 / D3) * sin(D3),
46                 (Dy3 / D3) * sin(D3),
47                 cos(D3)]])
48 t23 = Matrix([(L3 / D3**2) * Dx3 * (cos(D3) - 1),
49              (L3 / D3**2) * Dy3 * (cos(D3) - 1),
50              (L3 / D3**2) * D3 * sin(D3)])
51 t3 = t2 + R2 * t23
52 R3 = R2 * R23
53
54 t34 = Matrix([0, 0, L_gripper])
55 t3_gripper = t3 + R3 * t34
56 t3_total = Matrix.vstack(t3, t3_gripper)
57
58 variables_vector = Matrix([Dx1, Dy1, L1, Dx2, Dy2, L2, Dx3, Dy3, L3
59 ])
60 J = t3_total.jacobian(variables_vector)
61 return lambdify((Dx1, Dy1, L1, Dx2, Dy2, L2, Dx3, Dy3, L3),
62                t3_total, modules=['numpy']), lambdify((Dx1, Dy1, L1, Dx2, Dy2, L2,
63                Dx3, Dy3, L3), J, modules=['numpy']))
64
65 # Function that evaluates the configuration state starting from the
66 # cartesian coordinates using the jacobian matrix
67 def cartesian_configuration(new_joystick, old_joystick, result_J1,
68                            K_inv_original, DxDyL_new, DxDyL_old, limits, t3_c_result,
69                            threshold):
70     Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3 = DxDyL_new
71     e = new_joystick - old_joystick
72     K_inv = K_inv_original.copy()
73
74     # Evaluation of the jacobian matrix and solution to balance it with
75     # the stiffness matrix
76     J1 = result_J1(Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3)
77     J_transposed = J1.T

```

```

71 JK_inv = np.dot(J1, K_inv)
72 JK_inv_J_transposed = np.dot(JK_inv, J_transposed)
73 inv_JK_inv_J_transposed = np.linalg.pinv(JK_inv_J_transposed, rcond
=1e-5)
74 J_final = np.dot(K_inv, np.dot(J_transposed,
inv_JK_inv_J_transposed))
75
76 DxDyL_new += np.dot(J_final, e)
77 Dx1, Dy1, S1_original, Dx2, Dy2, S2_original, Dx3, Dy3, S3_original
= DxDyL_new
78
79 D1 = np.sqrt(Dx1**2 + Dy1**2)
80 D2 = np.sqrt(Dx2**2 + Dy2**2)
81 D3 = np.sqrt(Dx3**2 + Dy3**2)
82 print(f"\nDelta before adjusting: Dx1={Dx1}, Dy1={Dy1}, Dx2={Dx2},
Dy2={Dy2}, Dx3={Dx3}, Dy3={Dy3}, S1={S1_original}, S2={S2_original
}, S3={S3_original}")
83 eps = 1e-5
84
85 # Check of boundaries conditions and tune the stiffness matrix if
the solution has reached some bounds in terms of Delta
86 if D1 > limits[0]:
87     D1 = limits[0]
88     if np.linalg.norm(Dx1) > np.linalg.norm(Dy1):
89         K_inv[0,0] = K_inv[0,0]/100
90         Dx1 = np.sign(Dx1) * np.sqrt(D1**2 - Dy1**2)
91         if np.linalg.norm(Dy1) < eps:
92             Dy1 = np.sign(Dy1) * eps
93     else:
94         K_inv[1,1] = K_inv[1,1]/100
95         Dy1 = np.sign(Dy1) * np.sqrt(D1**2 - Dx1**2)
96         if np.linalg.norm(Dx1) < eps:
97             Dx1 = np.sign(Dx1) * eps
98 if D2 > limits[1]:
99     D2 = limits[1]
100    if np.linalg.norm(Dx2) > np.linalg.norm(Dy2):
101        K_inv[3,3] = K_inv[3,3]/100
102        Dx2 = np.sign(Dx2) * np.sqrt(D2**2 - Dy2**2)
103        if np.linalg.norm(Dy2) < eps:
104            Dy2 = np.sign(Dy2) * eps
105    else:
106        K_inv[4,4] = K_inv[4,4]/100
107        Dy2 = np.sign(Dy2) * np.sqrt(D2**2 - Dx2**2)
108        if np.linalg.norm(Dx2) < eps:
109            Dx2 = np.sign(Dx2) * eps
110 if D3 > limits[2]:
111     D3 = limits[2]
112     if np.linalg.norm(Dx3) > np.linalg.norm(Dy3):
113         K_inv[6,6] = K_inv[6,6]/100
114         Dx3 = np.sign(Dx3) * np.sqrt(D3**2 - Dy3**2)

```

```

115         if np.linalg.norm(Dy3) < eps:
116             Dy3 = np.sign(Dy3) * eps
117     else:
118         K_inv [7,7] = K_inv[7,7]/100
119         Dy3 = np.sign(Dy3) * np.sqrt(D3**2 - Dx3**2)
120         if np.linalg.norm(Dx3) < eps:
121             Dx3 = np.sign(Dx3) * eps
122
123     # Avoid numerical issues
124     if np.isnan(Dx1) or np.isnan(Dy1):
125         Dx1 = 0.000005
126         Dy1 = 0.000005
127         D1 = np.sqrt(Dx1**2 + Dy1**2)
128     if np.isnan(Dx2) or np.isnan(Dy2):
129         Dx2 = 0.000005
130         Dy2 = 0.000005
131         D2 = np.sqrt(Dx2**2 + Dy2**2)
132     if np.isnan(Dx3) or np.isnan(Dy3):
133         Dx3 = 0.000005
134         Dy3 = 0.000005
135         D3 = np.sqrt(Dx3**2 + Dy3**2)
136
137     # Adjust the length of each sections if the solution is outside the
138     # workspace defined
139     S1 = adjust_S(D1, S1_original, y_lower1, x_lower1_1, y_lower1_1,
140     y_upper1, x_upper1_1, y_upper1_1)
141     S2 = adjust_S(D2, S2_original, y_lower2, x_lower2_1, y_lower2_1,
142     y_upper2, x_upper2_1, y_upper2_1)
143     S3 = adjust_S(D3, S3_original, y_lower3, x_lower3_1, y_lower3_1,
144     y_upper3, x_upper3_1, y_upper3_1)
145     print(f"Delta: Dx1={Dx1}, Dy1={Dy1}, Dx2={Dx2}, Dy2={Dy2}, Dx3={Dx3}
146     }, Dy3={Dy3}, S1={S1}, S2={S2}, S3={S3}")
147
148     # Tune the stiffness matrix if the solution has reached some bounds
149     # in terms of S
150     if abs(S1 - S1_original) > 0.001:
151         K_inv [2,2] = K_inv[2,2]/100
152     if abs(S2 - S2_original) > 0.001:
153         K_inv [5,5] = K_inv[5,5]/100
154     if abs(S3 - S3_original) > 0.001:
155         K_inv [8,8] = K_inv[8,8]/100
156     DxDyL_new = np.array([Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3])
157
158     t3_c = t3_c_result (Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3).
159     reshape(-1)
160     print('goal:', new_joystick)
161     print('coordinates evaluated:', t3_c)
162     e = new_joystick - t3_c

```



```

157 # Loop to solve the IK if it's impossible to solve it with just 1
iteration
158 count = 0
159 while norm(e) > threshold:
160     count += 1
161     J1 = result_J1(Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3)
162     J_transposed = J1.T
163     JK_inv = np.dot(J1, K_inv)
164     JK_inv_J_transposed = np.dot(JK_inv, J_transposed)
165     inv_JK_inv_J_transposed = np.linalg.pinv(JK_inv_J_transposed,
rcond=1e-5)
166     J_final = np.dot(K_inv, np.dot(J_transposed,
inv_JK_inv_J_transposed))
167
168     DxDyL_new += np.dot(J_final, e)
169     Dx1, Dy1, S1_original, Dx2, Dy2, S2_original, Dx3, Dy3,
S3_original = DxDyL_new
170
171     D1 = np.sqrt(Dx1**2 + Dy1**2)
172     D2 = np.sqrt(Dx2**2 + Dy2**2)
173     D3 = np.sqrt(Dx3**2 + Dy3**2)
174     print("\nloop number", count)
175     print(f"Delta before adjusting: Dx1={Dx1}, Dy1={Dy1}, Dx2={Dx2
}, Dy2={Dy2}, Dx3={Dx3}, Dy3={Dy3}, S1={S1_original}, S2={
S2_original}, S3={S3_original}")
176     eps = 1e-5
177     if D1 > limits[0]:
178         D1 = limits[0]
179         if np.linalg.norm(Dx1) > np.linalg.norm(Dy1):
180             K_inv [0,0] = K_inv[0,0]/100
181             Dx1 = np.sign(Dx1) * np.sqrt(D1**2 - Dy1**2)
182             if np.linalg.norm(Dy1) < eps:
183                 Dy1 = np.sign(Dy1) * eps
184         else:
185             K_inv [1,1] = K_inv[1,1]/100
186             Dy1 = np.sign(Dy1) * np.sqrt(D1**2 - Dx1**2)
187             if np.linalg.norm(Dx1) < eps:
188                 Dx1 = np.sign(Dx1) * eps
189     if D2 > limits[1]:
190         D2 = limits[1]
191         if np.linalg.norm(Dx2) > np.linalg.norm(Dy2):
192             K_inv [3,3] = K_inv[3,3]/100
193             Dx2 = np.sign(Dx2) * np.sqrt(D2**2 - Dy2**2)
194             if np.linalg.norm(Dy2) < eps:
195                 Dy2 = np.sign(Dy2) * eps
196         else:
197             K_inv [4,4] = K_inv[4,4]/100
198             Dy2 = np.sign(Dy2) * np.sqrt(D2**2 - Dx2**2)
199             if np.linalg.norm(Dx2) < eps:
200                 Dx2 = np.sign(Dx2) * eps

```

```

201     if D3 > limits [2]:
202         D3 = limits [2]
203         if np.linalg.norm(Dx3) > np.linalg.norm(Dy3):
204             K_inv [6,6] = K_inv[6,6]/100
205             Dx3 = np.sign(Dx3) * np.sqrt(D3**2 - Dy3**2)
206             if np.linalg.norm(Dy3) < eps:
207                 Dy3 = np.sign(Dy3) * eps
208         else:
209             K_inv [7,7] = K_inv[7,7]/100
210             Dy3 = np.sign(Dy3) * np.sqrt(D3**2 - Dx3**2)
211             if np.linalg.norm(Dx3) < eps:
212                 Dx3 = np.sign(Dx3) * eps
213
214     if np.isnan(Dx1) or np.isnan(Dy1):
215         Dx1 = 0.000005
216         Dy1 = 0.000005
217         D1 = np.sqrt(Dx1**2 + Dy1**2)
218     if np.isnan(Dx2) or np.isnan(Dy2):
219         Dx2 = 0.000005
220         Dy2 = 0.000005
221         D2 = np.sqrt(Dx2**2 + Dy2**2)
222     if np.isnan(Dx3) or np.isnan(Dy3):
223         Dx3 = 0.000005
224         Dy3 = 0.000005
225         D3 = np.sqrt(Dx3**2 + Dy3**2)
226
227     S1 = adjust_S(D1, S1_original, y_lower1, x_lower1_1, y_lower1_1
228 , y_upper1, x_upper1_1, y_upper1_1)
229     S2 = adjust_S(D2, S2_original, y_lower2, x_lower2_1, y_lower2_1
230 , y_upper2, x_upper2_1, y_upper2_1)
231     S3 = adjust_S(D3, S3_original, y_lower3, x_lower3_1, y_lower3_1
232 , y_upper3, x_upper3_1, y_upper3_1)
233     print(f"Delta: Dx1={Dx1}, Dy1={Dy1}, Dx2={Dx2}, Dy2={Dy2}, Dx3
234 ={Dx3}, Dy3={Dy3}, S1={S1}, S2={S2}, S3={S3}")
235
236     if abs(S1 - S1_original) > 0.001:
237         K_inv [2,2] = K_inv[2,2]/100
238     if abs(S2 - S2_original) > 0.001:
239         K_inv [5,5] = K_inv[5,5]/100
240     if abs(S3 - S3_original) > 0.001:
241         K_inv [8,8] = K_inv[8,8]/100
242     DxDyL_new = np.array([Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3
243 ])
244
245     t3_c = t3_c_result (Dx1, Dy1, S1, Dx2, Dy2, S2, Dx3, Dy3, S3).
246 reshape(-1)
247     e = new_joystick - t3_c
248     print('goal:', new_joystick)
249     print('coordinates evaluated:', t3_c)

```

```

245     # Limits on number of iterations
246     if count == 10:
247         print("\nSATURATION OF ITERATIONS REACHED:", count)
248         DxDyL_new = DxDyL_old.copy()
249         t3_c = old_joystick.copy()
250         break
251     return DxDyL_new, t3_c
252
253 # Function that evaluates the tendon lengths starting from the
254 # configuration state
255 def configuration_tendon(DxDyL):
256     Dx1, Dy1, l_1, Dx2, Dy2, l_2, Dx3, Dy3, l_3 = DxDyL
257     Dx1 = -Dx1
258     Dy1 = -Dy1
259     Dx2 = -Dx2
260     Dy2 = -Dy2
261     Dx3 = -Dx3
262     Dy3 = -Dy3
263     D1 = np.sqrt(Dx1**2 + Dy1**2)
264     D2 = np.sqrt(Dx2**2 + Dy2**2)
265     D3 = np.sqrt(Dx3**2 + Dy3**2)
266     d = .070 #Tuned diameter of the base [m]
267
268     # Change of variables from Delta_x and Delta_y to theta and phi
269     theta1 = D1
270     theta2 = D2
271     theta3 = D3
272
273     if Dy1 > 0:
274         phi1 = np.arccos(Dx1/D1)
275     else:
276         phi1 = -np.arccos(Dx1/D1)
277     if Dy2 > 0:
278         phi2 = np.arccos(Dx2/D2)
279     else:
280         phi2 = -np.arccos(Dx2/D2)
281     if Dy3 > 0:
282         phi3 = np.arccos(Dx3/D3)
283     else:
284         phi3 = -np.arccos(Dx3/D3)
285
286     # Robot Kinematic
287     R1 = np.array ([[np.cos(phi1)**2*(np.cos(theta1)-1)+1, np.sin(phi1)*
288     np.cos(phi1)*(np.cos(theta1)-1), np.cos(phi1)*np.sin(theta1)],
289     [np.sin(phi1)*np.cos(phi1)*(np.cos(theta1)-1), np.
290     sin(phi1)**2*(np.cos(theta1)-1)+1, np.sin(phi1)*np.sin(theta1)],
291     [-np.cos(phi1)*np.sin(theta1), -np.sin(phi1)*np.sin(
292     theta1), np.cos(theta1)]])
293     R2 = np.array ([[np.cos(phi2)**2*(np.cos(theta2)-1)+1, np.sin(phi2)*
294     np.cos(phi2)*(np.cos(theta2)-1), np.cos(phi2)*np.sin(theta2)],

```

```

290         [np.sin(phi2)*np.cos(phi2)*(np.cos(theta2)-1), np.
sin(phi2)**2*(np.cos(theta2)-1)+1, np.sin(phi2)*np.sin(theta2)],
291         [-np.cos(phi2)*np.sin(theta2), -np.sin(phi2)*np.sin(
theta2), np.cos(theta2)]]
292
293     idx = np.linspace(0, 1, 40)
294     body1 = np.zeros((len(idx), 3))
295     rack1 = np.zeros((len(idx), 3))
296     rack2 = np.zeros((len(idx), 3))
297     rack3 = np.zeros((len(idx), 3))
298
299     #Offset to set in order to rotate the tendons
300     offset=np.radians(-90)
301     offset_sect2=np.radians(-30)
302     offset_sect3=np.radians(-30)
303
304     # Evaluation of cartesian coordinates along the tendons using
discrete steps
305     for m in range(len(idx)):
306         C1 = 1 - np.cos(idx[m]*theta1)
307         S1 = np.sin(idx[m]*theta1)
308         body1[m, :] = l_1/theta1 * np.array([np.cos(phi1)*C1, np.sin(
phi1)*C1, S1])
309
310         rack10 = np.array([d/2*np.cos(offset), d/2*np.sin(offset), 0])
311         rack1[m, :] = rack10 + (l_1 - theta1*rack10[0]*np.cos(phi1) -
theta1*rack10[1]*np.sin(phi1))/(theta1) * np.array([np.cos(phi1)*C1
, np.sin(phi1)*C1, S1])
312
313         rack20 = np.array([d/2*np.cos(2/3*np.pi+offset), d/2*np.sin
(2/3*np.pi+offset), 0])
314         rack2[m, :] = rack20 + (l_1 - theta1*rack20[0]*np.cos(phi1) -
theta1*rack20[1]*np.sin(phi1))/(theta1) * np.array([np.cos(phi1)*C1
, np.sin(phi1)*C1, S1])
315
316         rack30 = np.array([d/2*np.cos(4/3*np.pi+offset), d/2*np.sin
(4/3*np.pi+offset), 0])
317         rack3[m, :] = rack30 + (l_1 - theta1*rack30[0]*np.cos(phi1) -
theta1*rack30[1]*np.sin(phi1))/(theta1) * np.array([np.cos(phi1)*C1
, np.sin(phi1)*C1, S1])
318
319     body2 = np.zeros_like(body1)
320     rack4 = np.zeros_like(rack1)
321     rack5 = np.zeros_like(rack2)
322     rack6 = np.zeros_like(rack3)
323     for m in range(len(idx)):
324         C2 = 1 - np.cos(idx[m]*theta2)
325         S2 = np.sin(idx[m]*theta2)
326         body2[m, :] = l_2/theta2 * np.array([np.cos(phi2)*C2, np.sin(
phi2)*C2, S2])

```

```

327
328     rack40 = np.array([d/2*np.cos(offset+offset_sect2), d/2*np.sin(
offset+offset_sect2), 0])
329     rack4[m, :] = rack40 + (l_2 - theta2*rack40[0]*np.cos(phi2) -
theta2*rack40[1]*np.sin(phi2))/(theta2) * np.array([np.cos(phi2)*C2
, np.sin(phi2)*C2, S2])
330
331     rack50 = np.array([d/2*np.cos(2/3*np.pi+offset+offset_sect2), d
/2*np.sin(2/3*np.pi+offset+offset_sect2), 0])
332     rack5[m, :] = rack50 + (l_2 - theta2*rack50[0]*np.cos(phi2) -
theta2*rack50[1]*np.sin(phi2))/(theta2) * np.array([np.cos(phi2)*C2
, np.sin(phi2)*C2, S2])
333
334     rack60 = np.array([d/2*np.cos(4/3*np.pi+offset+offset_sect2), d
/2*np.sin(4/3*np.pi+offset+offset_sect2), 0])
335     rack6[m, :] = rack60 + (l_2 - theta2*rack60[0]*np.cos(phi2) -
theta2*rack60[1]*np.sin(phi2))/(theta2) * np.array([np.cos(phi2)*C2
, np.sin(phi2)*C2, S2])
336
337     body2[m, :] = (body1[-1, :] + np.dot(R1, body2[m,:]) ).T
338     rack4[m, :] = (body1[-1, :] + np.dot(R1, rack4[m,:]) ).T
339     rack5[m, :] = (body1[-1, :] + np.dot(R1, rack5[m,:]) ).T
340     rack6[m, :] = (body1[-1, :] + np.dot(R1, rack6[m,:]) ).T
341
342     body3 = np.zeros_like(body1)
343     rack7 = np.zeros_like(rack1)
344     rack8 = np.zeros_like(rack2)
345     rack9 = np.zeros_like(rack3)
346     for m in range(len(idx)):
347         C3 = 1 - np.cos(idx[m]*theta3)
348         S3 = np.sin(idx[m]*theta3)
349         body3[m, :] = l_3/theta3 * np.array([np.cos(phi3)*C3, np.sin(
phi3)*C3, S3])
350
351     rack70 = np.array([d/2*np.cos(offset+offset_sect2+offset_sect3)
, d/2*np.sin(offset+offset_sect2+offset_sect3), 0])
352     rack7[m, :] = rack70 + (l_3 - theta3*rack70[0]*np.cos(phi3) -
theta3*rack70[1]*np.sin(phi3))/(theta3) * np.array([np.cos(phi3)*C3
, np.sin(phi3)*C3, S3])
353
354     rack80 = np.array([d/2*np.cos(2/3*np.pi+offset+offset_sect2+
offset_sect3), d/2*np.sin(2/3*np.pi+offset+offset_sect2+
offset_sect3), 0])
355     rack8[m, :] = rack80 + (l_3 - theta3*rack80[0]*np.cos(phi3) -
theta3*rack80[1]*np.sin(phi3))/(theta3) * np.array([np.cos(phi3)*C3
, np.sin(phi3)*C3, S3])
356
357     rack90 = np.array([d/2*np.cos(4/3*np.pi+offset+offset_sect2+
offset_sect3), d/2*np.sin(4/3*np.pi+offset+offset_sect2+
offset_sect3), 0])

```

```

358     rack9[m, :] = rack90 + (l_3 - theta3*rack90[0]*np.cos(phi3) -
359     theta3*rack90[1]*np.sin(phi3))/(theta3) * np.array([np.cos(phi3)*C3
360     , np.sin(phi3)*C3, S3])
361
362     body3[m, :] = (body2[-1, :] + np.dot(R1, np.dot(R2, body3[m,:])
363     )).T
364     rack7[m, :] = (body2[-1, :] + np.dot(R1, np.dot(R2, rack7[m,:])
365     )).T
366     rack8[m, :] = (body2[-1, :] + np.dot(R1, np.dot(R2, rack8[m,:])
367     )).T
368     rack9[m, :] = (body2[-1, :] + np.dot(R1, np.dot(R2, rack9[m,:])
369     )).T
370
371     #Evaluation of the tendon lengths
372     l1=0
373     for m in range(len(rack1) - 1):
374         x1 = rack1[:, 0]
375         y1 = rack1[:, 1]
376         z1 = rack1[:, 2]
377         dx1 = x1[m+1] - x1[m]
378         dy1 = y1[m+1] - y1[m]
379         dz1 = z1[m+1] - z1[m]
380         l1 += np.sqrt(dx1**2 + dy1**2 + dz1**2)
381
382     l2=0
383     for m in range(len(rack2) - 1):
384         x2 = rack2[:, 0]
385         y2 = rack2[:, 1]
386         z2 = rack2[:, 2]
387         dx2 = x2[m+1] - x2[m]
388         dy2 = y2[m+1] - y2[m]
389         dz2 = z2[m+1] - z2[m]
390         l2 += np.sqrt(dx2**2 + dy2**2 + dz2**2)
391
392     l3=0
393     for m in range(len(rack3) - 1):
394         x3 = rack3[:, 0]
395         y3 = rack3[:, 1]
396         z3 = rack3[:, 2]
397         dx3 = x3[m+1] - x3[m]
398         dy3 = y3[m+1] - y3[m]
399         dz3 = z3[m+1] - z3[m]
400         l3 += np.sqrt(dx3**2 + dy3**2 + dz3**2)
401
402     l4=0
403     for m in range(len(rack4) - 1):
404         x4 = rack4[:, 0]
405         y4 = rack4[:, 1]
406         z4 = rack4[:, 2]
407         dx4 = x4[m+1] - x4[m]

```

```
402     dy4 = y4[m+1] - y4[m]
403     dz4 = z4[m+1] - z4[m]
404     l4 += np.sqrt(dx4**2 + dy4**2 + dz4**2)
405
406     l5=0
407     for m in range(len(rack5) - 1):
408         x5 = rack5[:, 0]
409         y5 = rack5[:, 1]
410         z5 = rack5[:, 2]
411         dx5 = x5[m+1] - x5[m]
412         dy5 = y5[m+1] - y5[m]
413         dz5 = z5[m+1] - z5[m]
414         l5 += np.sqrt(dx5**2 + dy5**2 + dz5**2)
415
416     l6=0
417     for m in range(len(rack6) - 1):
418         x6 = rack6[:, 0]
419         y6 = rack6[:, 1]
420         z6 = rack6[:, 2]
421         dx6 = x6[m+1] - x6[m]
422         dy6 = y6[m+1] - y6[m]
423         dz6 = z6[m+1] - z6[m]
424         l6 += np.sqrt(dx6**2 + dy6**2 + dz6**2)
425
426     l7=0
427     for m in range(len(rack7) - 1):
428         x7 = rack7[:, 0]
429         y7 = rack7[:, 1]
430         z7 = rack7[:, 2]
431         dx7 = x7[m+1] - x7[m]
432         dy7 = y7[m+1] - y7[m]
433         dz7 = z7[m+1] - z7[m]
434         l7 += np.sqrt(dx7**2 + dy7**2 + dz7**2)
435
436     l8=0
437     for m in range(len(rack8) - 1):
438         x8 = rack8[:, 0]
439         y8 = rack8[:, 1]
440         z8 = rack8[:, 2]
441         dx8 = x8[m+1] - x8[m]
442         dy8 = y8[m+1] - y8[m]
443         dz8 = z8[m+1] - z8[m]
444         l8 += np.sqrt(dx8**2 + dy8**2 + dz8**2)
445
446     l9=0
447     for m in range(len(rack9) - 1):
448         x9 = rack9[:, 0]
449         y9 = rack9[:, 1]
450         z9 = rack9[:, 2]
451         dx9 = x9[m+1] - x9[m]
```

```

452     dy9 = y9[m+1] - y9[m]
453     dz9 = z9[m+1] - z9[m]
454     l9 += np.sqrt(dx9**2 + dy9**2 + dz9**2)
455
456     L=np.array([l1 , l2 , l3 , l4 , l5 , l6 , l7 , l8 , l9])
457
458     return L
459
460 # Function to define the workspace for each section
461 def boundary_function(x, y1, x2, y2):
462     m = (y2 - y1) / x2
463     q = y1
464     return m * x + q
465
466 # Function to adjust the length of each section if the solution is
467     outside the workspace
468 def adjust_S(x, y, y_lower, x_lower_1, y_lower_1, y_upper, x_upper_1,
469     y_upper_1):
470     lower_limit = boundary_function(x, y_lower, x_lower_1, y_lower_1)
471     upper_limit = boundary_function(x, y_upper, x_upper_1, y_upper_1)
472     if y < lower_limit:
473         return lower_limit
474     elif y > upper_limit:
475         return upper_limit
476     else:
477         return y
478
479 Dx1, Dy1, L1, Dx2, Dy2, L2, Dx3, Dy3, L3 = symbols('Dx1 Dy1 L1 Dx2 Dy2
480     L2 Dx3 Dy3 L3')
481 l1, l2, l3, l4, l5, l6, l7, l8, l9 = symbols('l1 l2 l3 l4 l5 l6 l7 l8
482     l9 ')
483
484 # Limits for 1st section
485 y_lower1 = .055 #[m]
486 x_lower1_1, y_lower1_1 = np.pi/6, .065 #[rad], [m]
487 y_upper1 = .105 #[m]
488 x_upper1_1, y_upper1_1 = np.pi/6, .085 #[rad], [m]
489 # Limits for 2nd section
490 y_lower2 = .115 #[m]
491 x_lower2_1, y_lower2_1 = np.pi/3, .150 #[rad], [m]
492 y_upper2 = .255 #[m]
493 x_upper2_1, y_upper2_1 = np.pi/3, .150 #[rad], [m]
494 # Limits for 3rd section
495 y_lower3 = .125 #[m]
496 x_lower3_1, y_lower3_1 = np.pi/2, .165 #[rad], [m]
497 y_upper3 = .240 #[m]
498 x_upper3_1, y_upper3_1 = np.pi/2, .185 #[rad], [m]

```





# Appendix B

## Questionnaires

### B.1 Nasa TLX

#### **NASA Task Load Index**

*Hart and Staveland's NASA Task Load Index (TLX) method assesses work load on five 7-point scales. Increments of high, medium and low estimates for each point result in 21 gradations on the scales.*

Name	Task	Date
<p><b>Mental Demand</b>                      How mentally demanding was the task?</p>		
<p><b>Physical Demand</b>                      How physically demanding was the task?</p>		
<p><b>Temporal Demand</b>                      How hurried or rushed was the pace of the task?</p>		
<p><b>Performance</b>                      How successful were you in accomplishing what you were asked to do?</p>		
<p><b>Effort</b>                      How hard did you have to work to accomplish your level of performance?</p>		
<p><b>Frustration</b>                      How insecure, discouraged, irritated, stressed, and annoyed were you?</p>		

**Figure B.1:** Nasa Task Load Index.

## B.2 Trust in automation

### Checklist for Trust between People and Automation

Below is a list of statement for evaluating trust between people and automation. There are several scales for you to rate intensity of your feeling of trust, or your impression of the system while operating a machine. Please mark an "x" on each line at the point which best describes your feeling or your impression.

(Note: not at all=1; extremely=7)

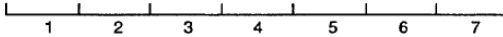
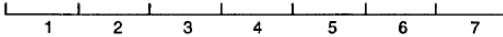
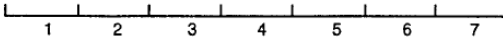
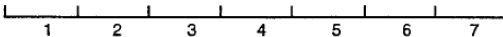
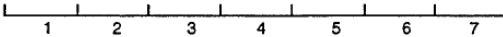
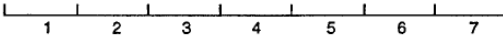
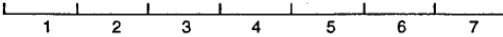
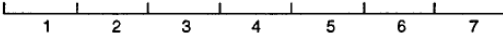
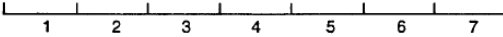
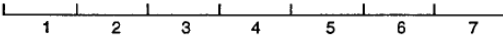
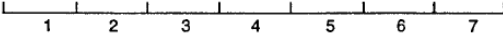
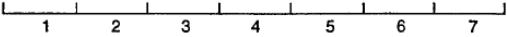
1	The system is deceptive	
2	The system behaves in an underhanded manner	
3	I am suspicious of the system's intent, action, or outputs	
4	I am wary of the system	
5	The system's actions will have a harmful or injurious outcome	
6	I am confident in the system	
7	The system provides security	
8	The system has integrity	
9	The system is dependable	
10	The system is reliable	
11	I can trust the system	
12	I am familiar with the system	

Figure B.2: Trust between people and automation.

# Bibliography

- [1] Deepak Trivedi, Christopher D. Rahn, William M. Kier, and Ian D. Walker. «Soft Robotics: Biological Inspiration, State of the Art, and Future Research». In: *Applied Bionics and Biomechanics* 5.3 (2008), p. 520417. DOI: <https://doi.org/10.1080/11762320802557865> (cit. on p. 2).
- [2] K. Suzumori, S. Iikura, and H. Tanaka. «Development of flexible microactuator and its applications to robotic mechanisms». In: *Proceedings. 1991 IEEE International Conference on Robotics and Automation*. 1991, 1622–1627 vol.2. DOI: 10.1109/ROBOT.1991.131850 (cit. on p. 3).
- [3] G. Robinson and J.B.C. Davies. «Continuum robots - a state of the art». In: *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*. Vol. 4. 1999, 2849–2854 vol.4. DOI: 10.1109/ROBOT.1999.774029 (cit. on pp. 3, 10).
- [4] C. Della Santina, Manuel G. Catalano, and Antonio Bicchi. «Soft Robots». In: *Encyclopedia of Robotics*. Feb. 2021. DOI: 10.1007/978-3-642-41610-1\_146-2 (cit. on p. 4).
- [5] Muhammad Muqet Rehman et al. Jahan Zeb Gul Memoon Sajid. «3D printing for soft robotics – a review». In: *Science and Technology of Advanced Materials* 19.1 (2018), pp. 243–262. DOI: 10.1080/14686996.2018.1431862 (cit. on p. 5).
- [6] François Schmitt, Olivier Piccin, Laurent Barbé, and Bernard Bayle. «Soft Robots Manufacturing : a Review». In: *Frontiers in Robotics and AI* 5 (2018), p. 84. DOI: 10.3389/frobt.2018.00084. URL: <https://hal.science/hal-03008794> (cit. on p. 5).
- [7] Michael A. Bell, Kaitlyn P. Becker, and Robert J. Wood. «Injection Molding of Soft Robots». In: *Advanced Materials Technologies* 7.1 (2022), p. 2100605. DOI: <https://doi.org/10.1002/admt.202100605> (cit. on p. 5).
- [8] Younan Xia and George M. Whitesides. «Soft Lithography». In: *Angewandte Chemie International Edition* 37.5 (1998), pp. 550–575. DOI: [https://doi.org/10.1002/\(SICI\)1521-3773\(19980316\)37:5<550::AID-ANIE550>3.0.CO;2-G](https://doi.org/10.1002/(SICI)1521-3773(19980316)37:5<550::AID-ANIE550>3.0.CO;2-G) (cit. on p. 5).

- [9] Weiqiang Dou, Guoliang Zhong, and Jinglin et. al. Cao. «Soft Robotic Manipulators: Designs, Actuation, Stiffness Tuning, and Sensing». In: *Advanced Materials Technologies* 6.9 (2021), p. 2100018. DOI: <https://doi.org/10.1002/admt.202100018> (cit. on pp. 6, 10, 11).
- [10] Xiaoqian Chen, Xiang Zhang, and Yiyong et. al. Huang. «A review of soft manipulator research, applications, and opportunities». In: *Journal of Field Robotics* 39.3 (2022), pp. 281–311. DOI: <https://doi.org/10.1002/rob.22051> (cit. on pp. 6, 10).
- [11] Daniela Rus and Michael Tolley. «Design, fabrication and control of soft robots». In: *Nature* 521 (May 2015), pp. 467–75. DOI: [10.1038/nature14543](https://doi.org/10.1038/nature14543) (cit. on p. 6).
- [12] B. Tondu and P. Lopez. «Modeling and control of McKibben artificial muscle robot actuators». In: *IEEE Control Systems Magazine* 20.2 (2000), pp. 15–38. DOI: [10.1109/37.833638](https://doi.org/10.1109/37.833638) (cit. on p. 6).
- [13] Jue Wang and Alex Chortos. «Control Strategies for Soft Robot Systems». In: *Advanced Intelligent Systems* 4 (Feb. 2022). DOI: [10.1002/aisy.202100165](https://doi.org/10.1002/aisy.202100165) (cit. on p. 7).
- [14] Chidanand Hegde, Jiangtao Su, Joel Ming Rui Tan, and He et al. «Sensing in Soft Robotics». In: *ACS Nano* 17.16 (2023). PMID: 37530475, pp. 15277–15307. DOI: [10.1021/acsnano.3c04089](https://doi.org/10.1021/acsnano.3c04089) (cit. on p. 8).
- [15] Michael Wehner and Michael T. et. al Tolley. «Pneumatic Energy Sources for Autonomous and Wearable Soft Robotics». In: *Soft Robotics* 1.4 (2014), pp. 263–274. DOI: [10.1089/soro.2014.0018](https://doi.org/10.1089/soro.2014.0018) (cit. on p. 8).
- [16] Turaj Ashuri and Ty et. al. Reasnor. «Biomedical soft robots: current status and perspective». In: *Biomedical Engineering Letters* 10 (May 2020), pp. 1–17. DOI: [10.1007/s13534-020-00157-6](https://doi.org/10.1007/s13534-020-00157-6) (cit. on p. 9).
- [17] Bryan Jones and I.D. Walker. «Kinematics for multisection continuum robots». In: *Robotics, IEEE Transactions on* 22 (Mar. 2006), pp. 43–55. DOI: [10.1109/TR0.2005.861458](https://doi.org/10.1109/TR0.2005.861458) (cit. on p. 11).
- [18] C. Della Santina, A. Bicchi, and D. Rus. «On an Improved State Parametrization for Soft Robots With Piecewise Constant Curvature and Its Use in Model Based Control». In: *IEEE Robotics and Automation Letters* 5 (Apr. 2020), pp. 1001–1008 (cit. on pp. 12, 50, 52).
- [19] Stanislao Grazioso, Giuseppe Di Gironimo, and Bruno Siciliano. «A Geometrically Exact Model for Soft Continuum Robots: The Finite Element Deformation Space Formulation». In: *Soft Robotics* 6.6 (2019), pp. 790–811. DOI: [10.1089/soro.2018.0047](https://doi.org/10.1089/soro.2018.0047) (cit. on p. 12).

- 
- [20] Della Santina C. et al. Qinghua G. Stella F. «Trimmed helicoids: an architected soft structure yielding soft robots with high precision, large workspace, and compliant interactions». In: *npj Robotics* 1.4 (Oct. 2023). DOI: 10.1038/s44182-023-00004-7 (cit. on pp. 13, 14).
- [21] Hokeun Kim, Yan Zhao, and Lihua Zhao. «Process-level modeling and simulation for HP’s Multi Jet Fusion 3D printing technology». In: *2016 1st International Workshop on Cyber-Physical Production Systems (CPPS)*. Apr. 2016, pp. 1–4. DOI: 10.1109/CPPS.2016.7483916 (cit. on p. 14).
- [22] Nilesh Sabnis. «A Review: State of The Art of Robotic Grippers». In: *International Journal of Engineering and Technology* (May 2018) (cit. on p. 19).
- [23] Josie Hughes, Utku Culha, and Fabio et. al. Giardina. «Soft Manipulators and Grippers: A Review». In: *Frontiers in Robotics and AI* 3 (2016). ISSN: 2296-9144. DOI: 10.3389/frobt.2016.00069 (cit. on p. 20).
- [24] Zhang Long, Qian Jiang, and Tao et. al. Shuai. «A Systematic Review and Meta-analysis of Robotic Gripper». In: *IOP Conference Series: Materials Science and Engineering* 782 (Apr. 2020), p. 042055. DOI: 10.1088/1757-899X/782/4/042055 (cit. on pp. 20, 21).
- [25] Jun Shintake, Vito Cacucciolo, Dario Floreano, and Herbert Shea. «Soft Robotic Grippers». In: *Advanced Materials* 30.29 (2018), p. 1707035. DOI: <https://doi.org/10.1002/adma.201707035> (cit. on p. 21).
- [26] Safvan Palathingal and G. Ananthasuresh. «Design of bistable arches by determining critical points in the force-displacement characteristic». In: *Mechanism and Machine Theory* 117 (July 2017), Pages 175–188. DOI: 10.1016/j.mechmachtheory.2017.07.009 (cit. on p. 22).
- [27] Dominic Vella Michael Gomez DerekE. Moulton. «Dynamics of viscoelastic snap-through». In: *Journal of the Mechanics and Physics of Solids* 124 (Mar. 2019), pp. 781–813 (cit. on pp. 23, 33, 34).
- [28] Nicholas P. Mastricola and Rajendra Singh. «Nonlinear load-deflection and stiffness characteristics of coned springs in four primary configurations». In: *Mechanism and Machine Theory* 116 (Oct. 2017), pp. 513–528 (cit. on pp. 23, 31).
- [29] J. O. Almen and A. Laszlo. «The Uniform-Section Disk Spring». In: *Transactions of the American Society of Mechanical Engineers* 58.4 (Feb. 2023), pp. 305–314. DOI: 10.1115/1.4020233 (cit. on p. 31).
- [30] Cosimo Della Santina, Giorgio Grioli, Manuel Catalano, Alberto Brando, and Antonio Bicchi. «Dexterity augmentation on a synergistic hand: The Pisa/IIT SoftHand+». In: *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. 2015, pp. 497–503. DOI: 10.1109/HUMANOIDS.2015.7363595 (cit. on p. 56).

- [31] «Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research». In: *Human Mental Workload*. Ed. by Peter A. Hancock and Najmedin Meshkati. Vol. 52. Advances in Psychology. North-Holland, 1988, pp. 139–183. DOI: [https://doi.org/10.1016/S0166-4115\(08\)62386-9](https://doi.org/10.1016/S0166-4115(08)62386-9) (cit. on p. 78).
- [32] Jiun-Yin Jian, Ann Bisantz, and Colin Drury. «Foundations for an Empirically Determined Scale of Trust in Automated Systems». In: *International Journal of Cognitive Ergonomics* 4 (Mar. 2000), pp. 53–71. DOI: [10.1207/S15327566IJCE0401\\_04](https://doi.org/10.1207/S15327566IJCE0401_04) (cit. on p. 81).