Design, modeling and implementation of a soft robotic neck for humanoid robots

by

Luis Fernando Nagua Cuenca

A dissertation submitted by in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering, Electronics and Automation

Universidad Carlos III de Madrid

Advisors:

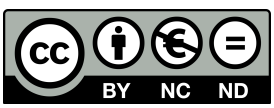
Concepción Alicia Monje Micharet

Carlos Balaguer Bernaldo de Quirós

Tutor:

Concepción Alicia Monje Micharet

June 2023



A mis padres Luis y Teresa y a mis hermanos Bryan y Marisol.

"Sometimes it is the people who no one imagines anything of who do the things no one can imagine."

"A veces las personas que todos piensan que no haran nada son las que hacen cosas que nadie habia imaginado."

Movie: The Imitation Game.

ACKNOWLEDGEMENTS

Thanks to God for giving me life and wisdom during these years.

To my parents for always supporting me and being with me all my life, to my siblings Marisol and Bryan for their support, but especially to my brother Bryan, because we have lived together day by day during these years. He has been a best friend. And of course thanks to my nieces Naty and Stefy and my brother-in-law Pablo.

My sincere thanks to Concha for the opportunity to work with her. It was the best experience I have ever had, and without her it would not have been possible. Thank you for sharing your knowledge with me and allowing me to learn about this wonderful world of research. And of course thank you for your friendship. I have got to know the wonderful person you are.

To the RoboticsLab group, especially to the HumaSoft group. They have been a family to me. I am grateful for their help even in the simplest things. A special mention to Jorge, Lisbeth, Carlos, Nicolle, Raúl and Natalia for their support and contribution to my thesis.

I would also like to thank David, Janeth, Dorin, Ángela, Edu and Sonia. They have been wonderful people and I really value their help and friendship. To Gyna and Dayana, who are spectacular people with whom it has been a pleasure to share these years of travelling and crazy experiences. Thanks also to Paul, Cristian, David and Segundo. Although I have only known them for a short time, they are people who have earned my appreciation and friendship. And finally, I would like to thank my childhood friends Daniel, Pancho, Omar and Cristian. Despite the distance, they continue to be unconditional friends.

In general, I am grateful to all the people I have met over these years. I will always carry you in my heart.

AGRADECIMIENTOS

Gracias a Dios por darme la vida y la sabiduría durante estos años.

A mis padres, por su apoyo en todo momento y por haberme acompañado a lo largo de mi vida, a mis hermanos Marisol y Bryan, por ser mi soporte, y especialmente a mi hermano Bryan, porque hemos vivido el día a día juntos durante estos años. Ha sido el mejor amigo. Por su puesto, a mis sobrinas Naty y Stefy y a mi cuñado Pablo.

Mi más sincero agradecimiento a Concha, por la oportunidad de trabajar junto a ella. Ha sido la mejor experiencia que he vivido. Sin ella no hubiera sido posible. Gracias por compartir tus conocimientos conmigo y por permitirme conocer el maravilloso mundo de la investigación. Y por supuesto, gracias por tu amistad, que me ha permitido conocer la maravillosa persona que eres.

Al grupo de investigación RoboticsLab, especialmente al grupo HumaSoft, que ha sido una verdadera familia para mí. Estoy agradecido por su ayuda hasta en las cosas más simples. Una mención especial a Jorge, Lisbeth, Carlos, Nicolle, Raúl y Natalia, cuyo apoyo y contribución a mi tesis ha sido muy grande.

También extiendo mis agradecimientos a David, Janeth, Dorin, Ángela, Edu y Sonia. Han sido unas personas maravillosas y me han brindado siempre su ayuda y su amistad. A Gyna y Dayana, por ser personas excelentes con quienes me ha sido grato compartir viajes y locuras durante estos años. Gracias también a Paul, Cristian, David y Segundo. Aunque los conozco desde hace poco, son personas que se han ganado mi aprecio y amistad. Y por último, quiero mostrar mi agradecimiento a mis amigos desde la niñez Daniel, Pancho, Omar y Cristian. A pesar de la distancia, siguen siendo amigos incondicionales.

En general estoy agradecido a todas las personas que he conocido en estos años. Siempre os llevare en mi corazón.

PUBLISHED AND SUBMITTED CONTENT

Journals:

1. J. Muñoz, C. A. Monje, **L. F. Nagua**, and C. Balaguer, "A graphical tuning method for fractional order controllers based on iso-slope phase curves," *ISA transactions*, vol. 105, pp. 296–307, 2020.

    - *The material of this source is partially included in this thesis in Chapter 6. The material from this source included in this thesis is not singled out with typographic means and references.*

2. L. Mena, C. A. Monje, **L. Nagua**, J. Muñoz, and C. Balaguer, "Test bench for evaluation of a soft robotic link," *Frontiers in Robotics and AI*, vol. 7, p. 27, 2020.

    - *The material of this source is partially included in this thesis in Chapter 6. The material from this source included in this thesis is not singled out with typographic means and references.*

3. **L. Nagua**, C. Relaño, C. A. Monje, and C. Balaguer, "A new approach of soft joint based on a cable-driven parallel mechanism for robotic applications," *Mathematics*, vol. 9, no. 13, p. 1468, 2021.

    - *The material of this source is partially included in this thesis in Chapter 3. The material from this source included in this thesis is not singled out with typographic means and references.*

4. N. A. Continelli, **L. F. Nagua**, C. A. Monje, and C. Balaguer, "Modelado de un cuello robótico blando mediante aprendizaje automático," *Revista Iberoamericana de Automática e Informática industrial*, 2023.

    - *The material of this source is partially included in this thesis in Chapters 5 and 6. The material from this source included in this thesis is not singled out with typographic means and references.*

Conferences:

1. **L. Nagua**, C. Monje, J. Muñoz, and C. Balaguer, "Design and performance validation of a cable-driven soft robotic neck," in *Jornadas Nacionales de Robótica*, 2018, pp. 1–5.

    - *The material of this source is partially included in this thesis in Chapters 3 and 4. The material from this source included in this thesis is not singled out with typographic means and references.*

2. **L. Nagua**, J. Muñoz, C. A. Monje, and C. Balaguer, "A first approach to a proposal of a soft robotic link acting as a neck," in *XXXIX Jornadas de Automática*, Área de Ingeniería de Sistemas y Automática, Universidad de Extremadura, 2018, pp. 522–529.

    - *The material of this source is partially included in this thesis in Chapters 3 and 4. The material from this source included in this thesis is not singled out with typographic means and references.*

3. L. Mena, C. A. Monje, **L. Nagua**, J. Muñoz, and C. Balaguer, "Sensorización de un sistema de eslabón blando actuando como cuello robótico," in *Jornadas Nacionales de Robótica*, vol. 6, 2019, pp. 98–102.

    - *The material of this source is partially included in this thesis in Chapter 6. The material from this source included in this thesis is not singled out with typographic means and references.*

4. **L. Nagua**, J. Muñoz, L. Mena, C. A. Monje, and C. Balaguer, "Robust control strategy for improving the performance of a soft robotic link," in *XLII Jornadas de Automática, Universidade da Coruña, Servizo de Publicacións*, 2021, pp. 499–506.

- *The material of this source is partially included in this thesis in Chapter 4. The material from this source included in this thesis is not singled out with typographic means and references.*

5. N. Continelli, **L. Nagua**, C. A. Monje, and C. Balaguer, "Identificación de un cuello robótico blando mediante aprendizaje automático," in *Jornadas de Robótica, Educación y Bioingeniería*, 2022, pp. 124–130.

    - *The material of this source is partially included in this thesis in Chapter 5. The material from this source included in this thesis is not singled out with typographic means and references.*

6. **L. Nagua**, N. Continelli, C. A. Monje, S. Martínez de la Casa, and C. Balaguer, "Machine learning techniques for the identification of a soft robotic neck," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2022, (Poster).

    - *The material of this source is partially included in this thesis in Chapters 5 and 6. The material from this source included in this thesis is not singled out with typographic means and references.*

OTHER RESEARCH MERITS

Patent:

- C. A. Monje, C. Relaño, L. Nagua, S. Martínez de la Casa, and C. Balaguer, “Eslabón para articulación blanda y articulación blanda que comprende dicho eslabón,” Ref. 5349ES, 2020.

Abstract

Soft humanoid robotics is an emerging field that combines the flexibility and safety of soft robotics with the form and functionality of humanoid robotics. This thesis explores the potential for collaboration between these two fields with a focus on the development of soft joints for the humanoid robot TEO. The aim is to improve the robot's adaptability and movement, which are essential for an efficient interaction with its environment.

The research described in this thesis involves the development of a simple and easily transportable soft robotic neck for the robot, based on a 2 Degree of Freedom (DOF) Cable Driven Parallel Mechanism (CDPM). For its final integration into TEO, the proposed design is later refined, resulting in an efficiently scaled prototype able to face significant payloads.

The nonlinear behaviour of the joints, due mainly to the elastic nature of their soft links, makes their modeling a challenging issue, which is addressed in this thesis from two perspectives: first, the direct and inverse kinematic models of the soft joints are analytically studied, based on CDPM mathematical models; second, a data-driven system identification is performed based on machine learning techniques. Both approaches are deeply studied and compared, both in simulation and experimentally.

In addition to the soft neck, this thesis also addresses the design and prototyping of a soft arm capable of handling external loads. The proposed design is also tendon-driven and has a morphology with two main bending configurations, which provides more versatility compared to the soft neck.

In summary, this work contributes to the growing field of soft humanoid robotics through the development of soft joints and their application to the humanoid robot TEO, showcasing the potential of soft robotics to improve the adaptability, flexibility, and safety of humanoid robots. The development of these soft joints is a significant achievement and the research presented in

this thesis paves the way for further exploration and development in this field.

Resumen

La robótica humanoide blanda es un campo emergente que combina la flexibilidad y seguridad de la robótica blanda con la forma y funcionalidad de la robótica humanoide. Esta tesis explora el potencial de colaboración entre estos dos campos centrándose en el desarrollo de una articulación blanda para el cuello del robot humanoide TEO. El objetivo es mejorar la adaptabilidad y el movimiento del robot, esenciales para una interacción eficaz con su entorno.

La investigación descrita en esta tesis consiste en el desarrollo de un prototipo sencillo y fácilmente transportable de cuello blando para el robot, basado en un mecanismo paralelo actuado por cable de 2 grados de libertad. Para su integración final en TEO, el diseño propuesto es posteriormente refinado, resultando en un prototipo eficientemente escalado capaz de manejar cargas significativas.

El comportamiemto no lineal de estas articulaciones, debido fundamentalmente a la naturaleza elástica de sus eslabones blandos, hacen de su modelado un gran reto, que en esta tesis se aborda desde dos perspectivas diferentes: primero, los modelos cinemáticos directo e inverso de las articulaciones blandas se estudian analíticamente, basándose en modelos matemáticos de mecanismos paralelos actuados por cable; segundo, se aborda el problema de la identificación del sistema mediante técnicas basadas en machine learning. Ambas propuestas se estudian y comparan en profundidad, tanto en simulación como experimentalmente.

Además del cuello blando, esta tesis también aborda el diseño de un brazo robótico blando capaz de manejar cargas externas. El diseño propuesto está igualmente basado en accionamiento por tendones y tiene una morfología con dos configuraciones principales de flexión, lo que proporciona una mayor versatilidad en comparación con el cuello robótico blando.

En resumen, este trabajo contribuye al creciente campo de la robótica humanoide blanda mediante el desarrollo de articulaciones blandas y su aplicación al robot humanoide TEO,

mostrando el potencial de la robótica blanda para mejorar la adaptabilidad, flexibilidad y seguridad de los robots humanoides. El desarrollo de estas articulaciones es una contribución significativa y la investigación presentada en esta tesis allana el camino hacia nuevos desarrollos y retos en este campo.

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

1.1 Motivation

Robotics is one of the most fascinating areas of research and technological development in recent times. From the creation of the first robots in the 1960s to the present day, robotics has advanced by leaps and bounds, enabling the creation of increasingly complex and sophisticated robots. One of the most interesting areas of robotics is humanoid robotics, which focuses on creating robots with human like characteristics. Throughout history, humanoid robots have been an obsession for many scientists and futurists, who have imagined worlds in which robots coexist naturally with humans. In this context, soft robotics offers a new opportunity to develop robots with more human like and flexible characteristics.

The idea of creating robots with human characteristics dates back to ancient times. The first prototypes of humanoid robots were developed in the 20th century. In 1961, the US company Unimation unveiled the first industrial robot with human-like arms and hands, called Unimate. Since then, a number of humanoid robot projects and prototypes have emerged, including the Japanese WABOT-1 in 1972, ASIMO by Honda in 2000 and Atlas by Boston Dynamics in 2013. These advances have allowed robots to increasingly approach the complexity and flexibility of the human body. However, there is still much work to be done before humanoid robots become an everyday reality in our society.

While the progress in robotics over the last 50 years is undeniable, robots made of rigid materials still have many limitations. Today, there is a new wave of robots inspired by biological designs with "soft" elements that are capable of performing tasks that are beyond the reach of rigid-limbed robots.

In this context, soft robotics presents a new opportunity to develop robots with more human-like and flexible characteristics. This new paradigm is known as Soft Robotics and is presented as an innovation beyond the already known Variable Stiffness Actuator (VSA) robots and flexible robots, incorporating soft links to robots. Indeed, the euRobotics MAR (Multi-Annual Roadmap) identifies "soft robotics design" as one of the most important interdisciplinary challenges in modern robotics.

In the case of humanoid robotics, a soft-link robot has the following main advantages over a rigid design:

- Simplicity of design, with a sub-actuated architecture without the need to increase the number of degrees of freedom.
- Increased accessibility and adaptability to complex environments, with postural control that is difficult to implement in rigid robots.
- Safer interaction with people and the environment, with high levels of shock absorption and increased stability.

In view of the above, the researcher Concepción A. Monje proposes the HumaSoft project (Diseño y Control de Eslabones Blandos para Robots Humanoides (HumaSoft), Ref. DPI2016-75330-P, Ministry of Economy, Industry and Competitiveness, 30/12/2016-31/12/2020), whose objective is to develop a new type of joint to be incorporated into humanoid robots, with the characteristics of simplicity, accessibility and safety.

As part of the HumaSoft project, this thesis aims to develop the design and implementation of a soft robotic link that can be incorporated into the humanoid robot TEO (Task Environment Operator), created by the RoboticsLab research group of the Universidad Carlos III de Madrid. The main objective of the project is to develop a soft robotic link that allows the robot to better adapt to its environment and its tasks. The design and development of the joint will be based on the principles of soft robotics to create a flexible, adaptable and safe component for its use in the humanoid robot TEO.

1.2 Objectives

The main objective of this thesis is to develop a new type of soft joint that meets the key characteristics of simplicity, accessibility and safety. This link will be designed, implemented

and validated to achieve optimal and safe functionality. Once developed, this soft joint will be incorporated into the humanoid robot TEO, enhancing its ability to perform complex tasks and allowing for greater interaction with humans.

A number of sub-objectives are proposed as a road map for the achievement of the main objective:

- Review of the state of the art in soft robotics for humanoid robots.
- Design and development of a soft-link prototype, defining the material and actuation system that will act as the robotic neck.
- Develop the analytical model of the soft robotic neck.
- Use artificial intelligence techniques to identify the kinematic model of the soft robotic neck.
- Re-design of the soft link prototype to be easily integrated into the structure of the humanoid robot TEO.
- Perform simulation tests and real experiments to validate the performance of the soft robotic neck.
- Analyse and validate the experiments carried out on the soft robotic neck.

1.3 Document organization

This document is organised into seven different chapters, including this Introduction chapter. The main topics addressed in these chapters are:

- Chapter 2 presents an overview of the state of the art of soft robotics and focuses on the evolution of the technology by using flexible materials to build more elastic, lighter and stronger robots. In addition, soft robotics is presented as a solution for the development of humanoid robots.
- Chapter 3 proposes a cable-driven soft robotic mechanism to act as a neck, defining its material and actuation system. This is followed by a redesign of the soft neck prototype with the goal of incorporating it into the humanoid robot TEO. Finally, a design for a soft

robotic arm is presented, focusing on the versatility of the soft joint to be used in different parts of the humanoid robot.

- Chapter 4 presents a detailed description of the mathematical models of the cable-driven parallel mechanisms described in Chapter 2. In particular, for the spring-based neck and the soft link-based neck, the inverse kinematics and direct kinematics are analysed. In addition, the validation and simulation of the models are discussed, which makes it possible to evaluate their performance.

- Chapter 5 presents a machine learning approach applied to the problem of identifying the inverse kinematics of the soft robotic neck. A multi-layer perceptron neural network is used to train an identification system that establishes a relationship between the desired neck positions and the angular configuration of the three motors acting as actuators. The effectiveness of the proposed architectures are evaluated by simulations.

- Chapter 6 presents the results of the experiments carried out on the soft neck prototypes. Individual tests are carried out on each of the prototypes to validate their models and performance.

- Finally, Chapter 7 outlines the main conclusions of the thesis and some future works.

Chapter 2

STATE OF THE ART

The 20th century saw an unprecedented technological revolution at breakneck speed, with the advent of robotics at the end of the century, which produced designs inspired by the biology of the animal kingdom and, of course, human biology. However, robotics in this period used rigid materials to structure their mechanisms and were therefore heavy, which affected their mobility. Soft robotics is designed to overcome the problems of conventional robotics. This is the term used to describe robotic systems that use flexible materials instead of rigid plastics, metal or other similar materials to build them. The result is a flexible, elastic robot that can withstand drops and bumps. It can compress to fit into tight spaces and perform more complex and natural movements.

The study of soft robotics began in the area of biology, with the goal of imitating the behaviour of some invertebrate animals in order to develop flexible and resistant robots. This was reflected in early soft robotics research, such as that of the famous MIT research team, which in 2008 developed the first octopus-inspired soft robot (Figure 2.1), called Octobot [1]. This research has opened the door to a new age of robotics, where robots can adapt to any situation and manoeuvre with flexibility and precision.

An overview of the main components of soft robotics is provided in this chapter. First of all, it classifies and discusses the most commonly used materials for soft robots, with a special focus on 3D printing due to its increasing importance in the manufacture of soft components. Later, the main actuators used for the actuation of these types of robots are discussed. Finally, an insight into Humanoid Soft Robotics is given, which has become an emerging field of research in recent years. This technology seeks to replicate the shape and behaviour of human beings. These humanoid robots are designed to replicate human anatomy, movements and other skills

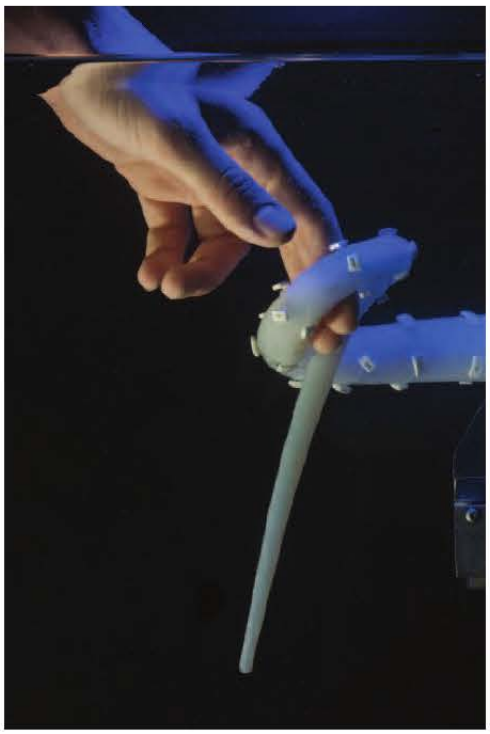


Figure 2.1: The first octopus-inspired soft robot [1].

needed to interact with the environment.

2.1 Materials in soft robotics

Materials used in soft robotics are those that can deform and adapt to different environments, such as rubbers, fabrics, foams, membranes and other flexible structures. These materials enable the development of devices that are safer, softer, more shock-resistant and better adapted to physical environments [2]. This is because soft materials have properties such as elasticity, resilience, abrasion resistance and the ability to adapt to different shapes. These materials are becoming increasingly important for the creation of robots able to work in difficult-to-access environments.

Table 2.1 [2]–[6] presents the main materials and their respective properties that are used in soft robotics.

2.1.1 Silicone

Silicone is an ideal material for soft robotics due to its high resistance to traction, good elasticity and temperature resistance. In addition, its versatility and wear resistance allows it

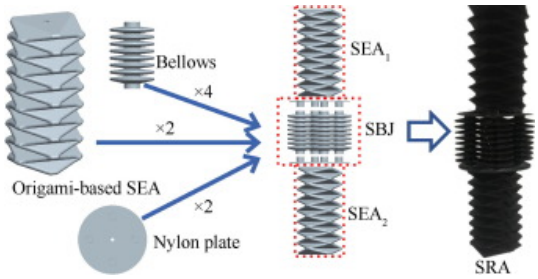
Table 2.1: The main materials used in soft robotics.

Material	Relevant Properties
Silicone	Durable, Flexible, High Tensile Strength
Polyurethane	Water Resistant, Chemically Resistant, Lightweight
Neoprene	Flexible, Resistant to Abrasion, Good Insulation
Nylon	Lightweight, Strong, Resistant to UV Light
Polyester	Good Strength, Low Cost, Easily Moldable

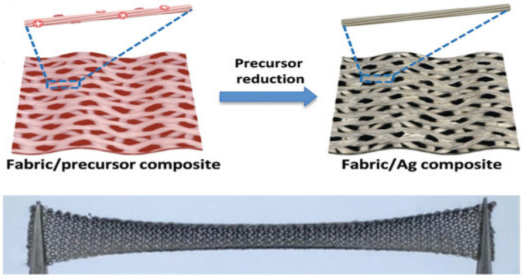
(a) Pneumatics made of silicone.

(b) Hand made of polyurethane.

(c) Fingers made of neoprene.



(d) Soft robotic arm made of nylon.



(e) Textile made of polyester.

Figure 2.2: Materials in soft robotics.

to make flexible and resistant robots [7]. This is due to both its elastic properties and its low hardness, which allows it to be moulded into more complex forms. Silicone is also a water-resistant material, which facilitates its use in humid environments [8], [9]. In addition, its soft surface allows for better interaction between robots and human environments [10]. For all these

reasons, silicone has become one of the main materials used in the manufacture of soft robots (Figure 2.2a).

2.1.2 Polyurethane

Polyurethane (PU) is an increasing popular material in soft robotics. It is flexible, strong, lightweight and inexpensive. According to [11], [12], this material has been used as a framework and cover material for many soft robots because of its relatively low cost, good abrasion and wear resistance, flexibility and compressive strength.

Polyurethane has been tested in different additive manufacturing processes, including 3D printing, fused deposition modeling, direct ink writing, selective laser sintering, inkjet printing and stereolithography (Figure 2.2b). In addition, it has been shown to be a suitable material for the manufacture of environmentally compatible robots, due to its low toxicity and easy degradability [13].

2.1.3 Neoprene

Neoprene, also known as polychloroprene, is a synthetic rubber material with a wide range of applications, including soft robotics [14]. Neoprene is an elastic material, making it ideal for use in soft-bodied robots, as it can be used to create a range of flexible, deformable structures (Figure 2.2c). It is also highly resistant to temperatures, UV radiation, and many chemicals, which makes it well-suited for use in soft robots operating in harsh environments [5]. Additionally, neoprene can be used to create air-filled chambers and chambers filled with other liquids, allowing for the creation of pneumatic or hydraulic actuators [14], [15]. Finally, neoprene offers the potential for high performance due to its damping properties, durability, and stiffness.

2.1.4 Nylon

Nylon is an ideal material for soft robotics due to its robustness, low cost and versatility. Its low elastic modulus and high strength make it well suited for producing soft robotic structures with high levels of flexibility and precision [16]. Nylon can also be used for the fabrication of actuators and sensors, which can improve the control and accuracy of soft robotic systems [17] (Figure 2.2d). The low thermal conductivity of nylon makes it ideal for insulation, and its non-conductive properties make it suitable for use in various types of electricity-generating robots

[18]. In addition, nylon is highly wear resistant, making it a durable material for long-term use [19]. All of these properties make nylon an excellent choice for soft robotic applications.

2.1.5 Polyester

Soft robotics is a rapidly growing field and polyester is an increasingly popular material for use in soft robotic applications. Polyester has a number of advantageous properties for soft robotics, including light weight, durability, flexibility, and excellent abrasion and tear resistance, making it an ideal material for use in the manufacture of soft actuators, sensors and other soft robotic components [20], [21] (Figure 2.2e). In addition to its applications in soft robotics, polyester is also used in other areas of robotics. For example, polyester fabrics are used as structural materials in robots, as well as for insulation and damping. Polyester is also used to make sensors, which are important for detecting and responding to environmental changes [22].

2.2 Actuators in soft robotics

The soft robots are capable of performing delicate tasks, and their deformable bodies allow them to move in ways that traditional robots cannot. Actuators are the components of soft robots that allow them to move and interact with their environment [23]. Soft actuators are typically made of materials such as shape memory alloys, dielectric elastomers and electroactive polymers [2]. These materials have properties such as contractility, deformability and flexibility that allow the robot to move and interact with its environment in a much more precise and compliant way than traditional robots.

2.2.1 Soft pneumatic actuators

Soft pneumatic actuators (SPA) are a type of soft robotic actuator that use pressurised air or gas to generate force and motion. They are made of a flexible material such as silicone rubber and are designed to be highly compliant and responsive to changes in pressure [24]. They can be used to create simple movements such as pushing and pulling, or to control more complex movements such as walking and swimming. Soft pneumatic actuators are ideal for use in soft robots due to their light weight, low cost and flexibility. They can also be used to create soft gripping tools, medical instruments and prosthetic limbs [25].

2.2.1.1 Pneumatic networks

The actuator can be divided into two layers: an extendable upper layer and a restricted lower layer. The top layer consists of a linear array of air chambers connected internally by an air channel. Figure 2.3 illustrates the structure of a pneumatic network (Pneu-net). When inflated, the inner walls of the air chambers of the upper layer expand and compress each other, while the lower layer limits the elongation in the longitudinal direction. This creates a bending movement of the entire actuator to the side of the restrictive layer.

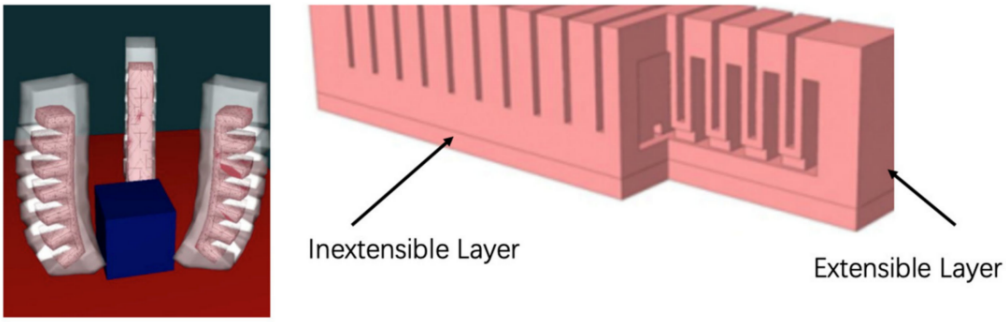


Figure 2.3: Pneumatic soft actuators structure [24].

Pneu-net can also be used to produce deformations other than bending. By changing the angle of the chambers, a combination of bending and twisting can be achieved, as shown in Figure 2.4.

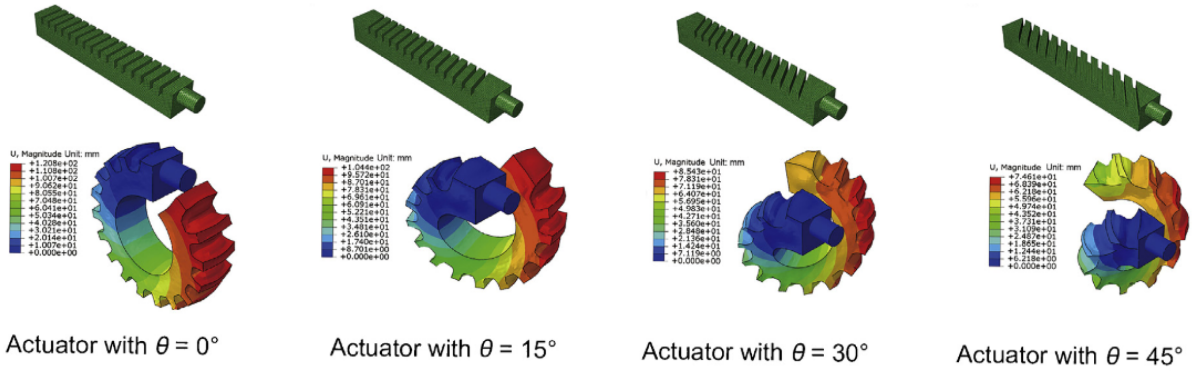


Figure 2.4: Pneu-net actuator with bending plus twisting motion [26].

Flexible Fluidic Actuator (FFA) is an extensible and adaptive soft actuator composed of elastomers and multiple channels embedded within them. These channels can be inflated and deformed when exposed to fluid pressure (Figure 2.5), allowing for adaptive flexibility and movement. This type of actuator is inspired by biological structures and can be used to replicate their movements. The flexibility of the FFA also makes it ideal for use in medical applications

such as prosthesis, tissue engineering, and drug delivery [27].

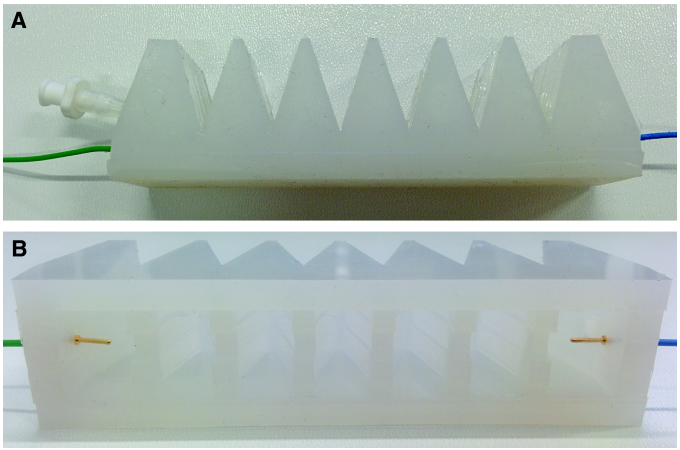


Figure 2.5: (A) A proprioceptive bending FFA. (B) The proprioceptive bending FFA part way through fabrication, showing the electrodes necesssary to measure the resistance of the conductive fluid [28].

Honeycomb Pneumatic Networks are increasingly being used in soft robotics applications due to their ability to provide relatively rigid structures with high energy absorption and low stiffness. The honeycomb network consists of a series of interconnected hexagonal cells filled with a polymer or foam material (Figure 2.6). This structure provides several benefits for soft robots, including improved stiffness, flexibility and shock absorption [24]. In addition, the honeycomb network provides a low-friction surface that can be used to improve the performance of robotic actuators and reduce the energy required to move them. The honeycomb structure can also be used to create complex shapes, such as curved and contoured surfaces. This is useful for creating more dynamic and adaptive robots that can navigate complex environments [29].

2.2.1.2 Fibre-constrained actuators

The main feature of fibre-constrained pneumatic flexible actuators is their ability to generate anisotropic mechanical behaviour when air pressure is applied, due to the presence of fibres, fabrics or other fibre-like structures that reinforce the elastic cavity. This anisotropic motion gives these actuators a unique set of properties, such as high stiffness and large contraction ratios, which can be used in a variety of applications [30], [31]. The pneumatic artificial muscle (PAM) is the best known example of this type of actuator.

PAMs are made of elastomeric materials such as rubber or silicone that are inflated with air or other gases like the one shown in Figure 2.7. They can generate large forces and strokes while remaining lightweight, flexible and inexpensive. PAMs can be used to create robotic arms,

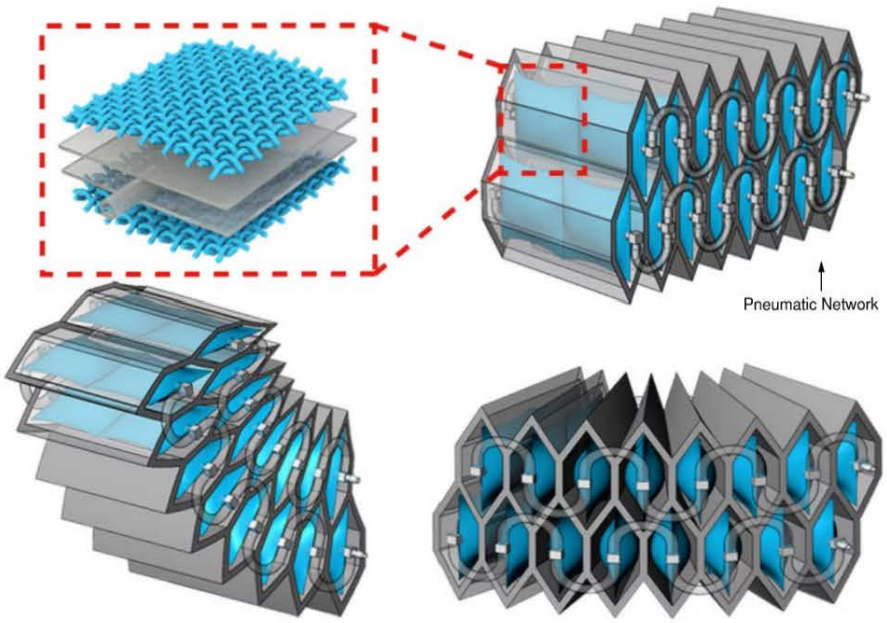


Figure 2.6: The honeycomb pneumatic networks arm [24].

grippers and other robotic devices that can mimic the movement of biological muscles [32]. They are also used in medical applications, such as robotic prosthetics and surgical instruments. PAMs are seen as a promising technology for soft robotics, as they can provide a more intuitive and natural way of controlling robots.

Compared to other actuation mechanisms (electric motors, hydraulic actuators, and shape memory alloys), PAMs offer several advantages [32]–[35]:

1. Low weight: PAMs are often much lighter than other actuation mechanisms due to their simple design and the use of lightweight materials such as carbon fiber.

2. High strength: PAMs are capable of generating large amounts of force in a relatively small package, making them ideal for applications that require a lot of force in a small space.

3. Compact size: PAMs are small and lightweight, making them ideal for applications that require a compact actuation mechanism.

4. Low cost: PAMs are typically much less expensive than other actuation mechanisms, making them a cost-effective solution for many applications.

5. Low maintenance: PAMs require very little maintenance, making them an attractive option for applications that do not require frequent maintenance or repair.

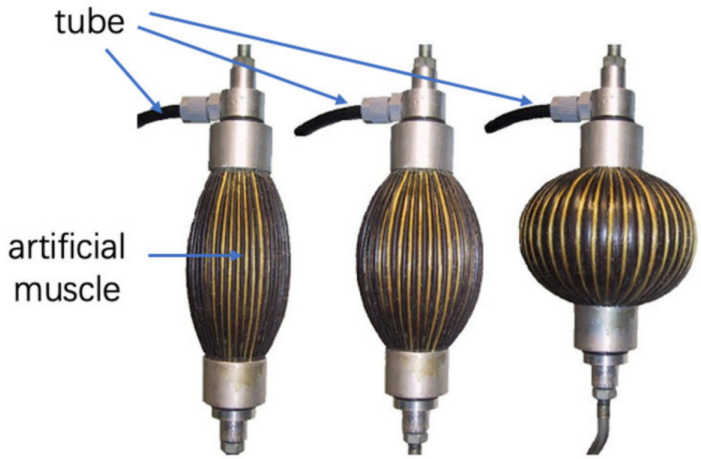


Figure 2.7: McKibben artificial muscle [24].

2.2.2 Vacuum actuators

Vacuum actuators exploit the same principles of asymmetry discussed before. In contrast to pneumatic actuation, negative pressure is utilized for actuation (Figure 2.8). For instance, a three-chamber actuator driven by negative pressure is featured in [36] (Figure 2.9a). Linear muscles which contract [37] (Figure 2.9b) and origami structures [38] (Figure 2.9c) are other examples of vacuum actuators.

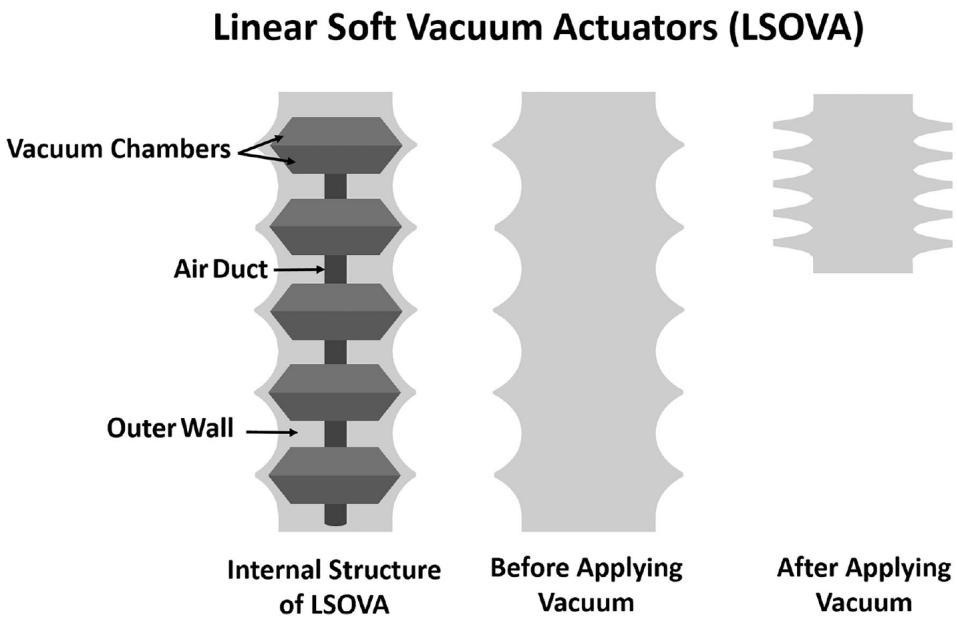
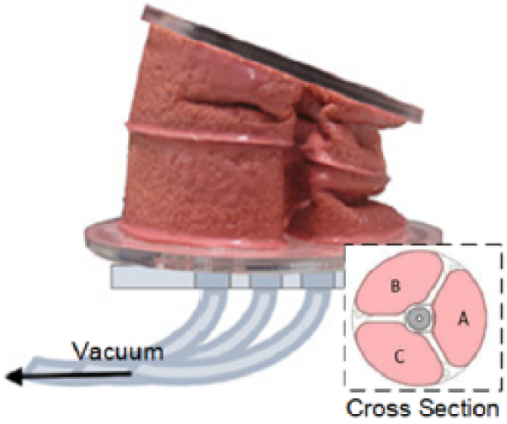
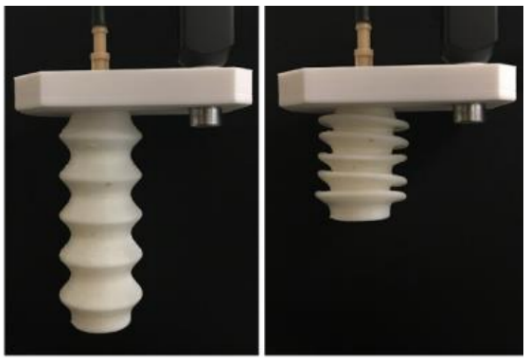


Figure 2.8: Working principle of vacuum actuation [39].

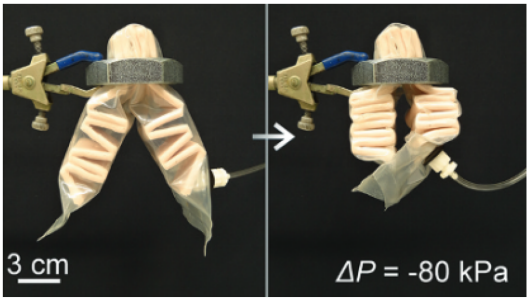
Authors in [40] designed and fabricated vacuum bending actuators (VBAs) for use in continuum manipulators. To evaluate their effectiveness, they tested a single VBA and a bimorph actuator consisting of two VBAs to achieve multi-plane motion as shown in Figure 2.9d. Finally,



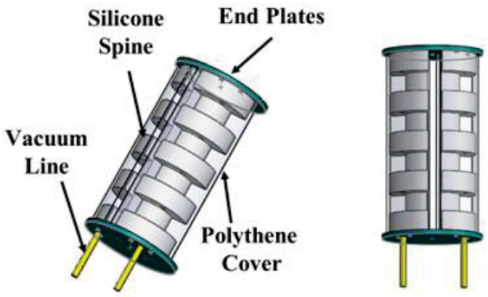
(a) Three-chamber vacuum bending actuator [36].



(b) Contracting muscle based on a patterned structure [37].



(c) Origami vacuum actuator [38].



(d) Bimorph created with two bending vacuum actuators [40].

Figure 2.9: Examples of vacuum actuators.

they constructed a continuum manipulator with three VBAs and demonstrated its successful operation.

2.2.3 Cable-driven (tendon-driven) actuators

Cable-driven or tendon-driven soft robots feature actuators situated outside of the robot structure, allowing their movements to remain unrestricted. The actuators are connected to the robot via cables, which transmit the tension through cable routes integrated in the soft structure [41]. Nonlinearity and hysteresis are major problems in the control of soft robots using cable-driven mechanisms. High friction between a cable and cable path can cause a variety of problems, such as tension of the cable leading to increased wear and tear, and bending of the cable path causing the cable to rub against the side walls [42]. These problem can be resolved by reducing the tension of the cable, smoothing out the bends in the cable path, and using low-friction materials, such as polytetrafluoroethylene (PTFE) to reduce the friction between the

cable and the cable path [43].

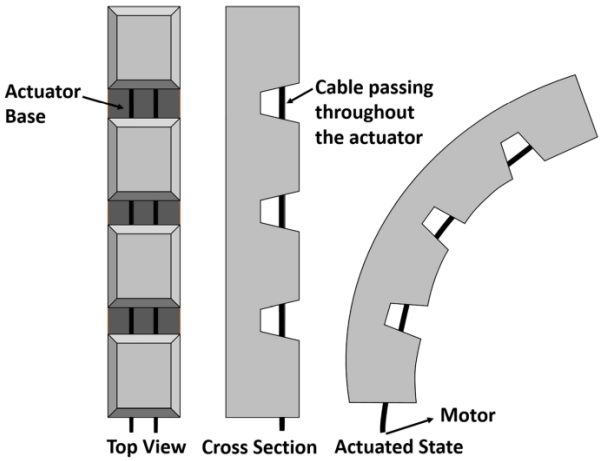


Figure 2.10: Cable-driven actuation [39].

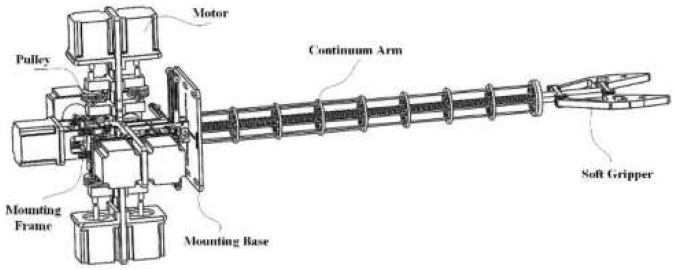
The cable actuation mechanism is a popular choice for soft robotic grippers and manipulators [39]. This method works by controlling the movement of the soft body by retracting the cables, which are embedded in the structure and anchored at specific points (Figure 2.10). The design of the structure presents a challenge, as it requires careful placement of motors, pulleys, force sensors and encoders.

The work in [44] implemented this actuation mechanism in a two-sectioned continuum arm, driven by eight cables, each section containing four cables as shown in Figure 2.11b. The arm was equipped with a cable-driven soft gripper, enabling it to perform various picking operations (Figure 2.11a). Continuum soft robots also use tendon actuation, like the tentacle in [45], [46] (Figure 2.12).

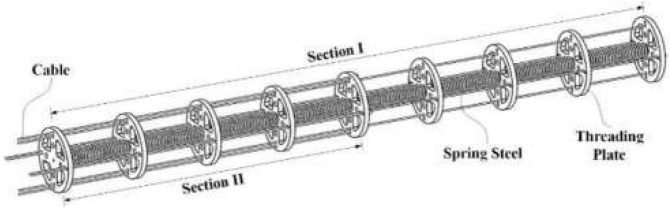
Due to their characteristics and properties, cable-driven actuators are used in the soft robotic joints proposed in this thesis, as described in Chapter 3.

2.2.4 Electrical actuators

Pneumatic and tendon-based actuators are the predominant choices in the field of soft robotics due to their strength and versatility. However, electrical actuators are becoming increasingly attractive for use in this field due to their ease of control and ability to generate a wide range of forces. Electrical actuators also have the advantage of being relatively quiet and suitable for use in environments that involve humans. Therefore, soft electrical actuators are increasingly being studied and investigated for their potential applications in soft robotics [47].



(a) General design schematic.



(b) Design of cable-driven continuum manipulator.

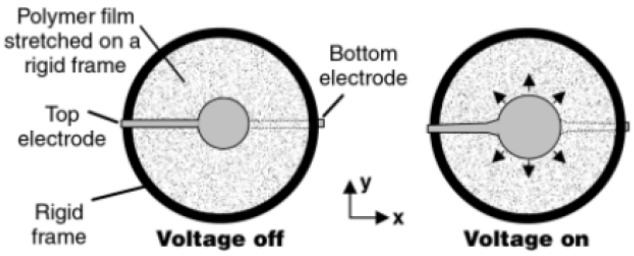
Figure 2.11: The soft gripper [44].

Figure 2.12: Stretch motion of cable-driven bionic tentacles [46].

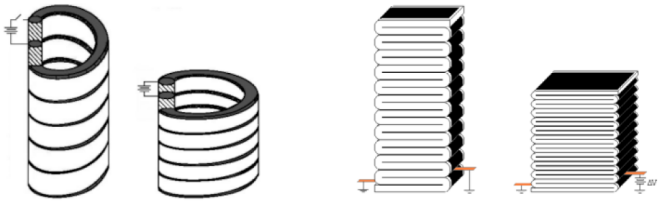
A truly soft electrical-powered actuator can be made of dielectric elastomers, as presented in [48], where the most widespread concept of a dielectric elastomer actuator (DEA) was introduced: a DE film between two electrodes, which can achieve area expansions of more than 100% when actuated with large electric fields (Figure 2.13a). This effect was used in [49] to create DE linear muscles with helical and folded designs (Figure 2.13b). Dielectric elastomers offer an alternative to fluidic actuators, but their application in soft robotics has been limited by the need for high voltages.

The same idea, but using shape memory alloys (SMA) have been used in a variety of artificial muscles, such as the octopus-like tentacle found in [50] (Figure 2.14). Another application of electric-powered SMA-based actuators is a worm-like robot, as described in [51].

SMA have also many advantages for use in soft grippers, such as low noise, high force to weight ratio, small size [52], and ease of use (Figure 2.15).



(a) Basic DEA [48].



(b) Helical and folded DE linear muscle [49].

Figure 2.13: Different DEA designs.

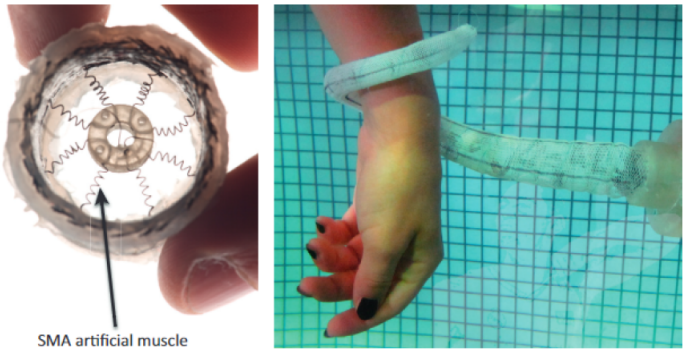


Figure 2.14: SMA-based artificial muscle and tentacle robot [50].

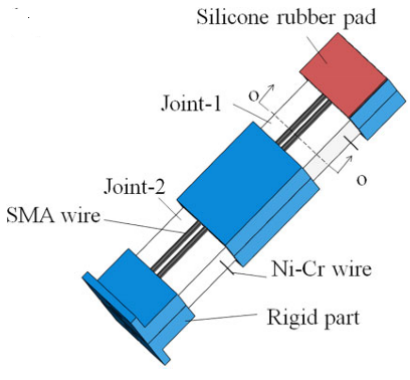


Figure 2.15: Design of a soft robotic finger [52].

2.3 Soft robotics applied to humanoid robots

Since around 2010, research on humanoid soft robots, soft grippers, soft actuators and soft sensors has rapidly emerged and been actively studied (Figure 2.16). The history of research on

bipedal humanoid robots goes back to the 1970s and became particularly active after the 2000s. The publication record for humanoid robots peaked around 2015. Since then, the publication record shows a decrease in research on humanoid robots, while research on soft robotics has increased consistently since around 2019 [53].

This thesis explores the potential for collaboration between humanoid robots and soft robotics, leading to the emergence of a new field of study: soft humanoid robotics. By combining their respective histories and publication trends, these two fields of robotics have the potential to create a novel approach to robotics that could have many applications.

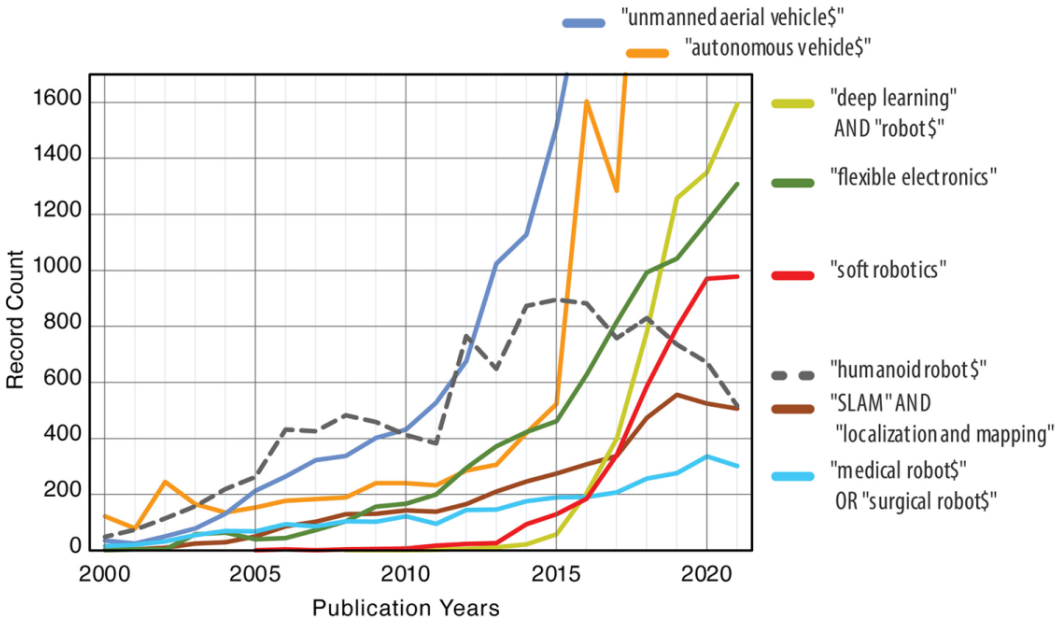


Figure 2.16: Number of publications per research topic based on keyword searches on the Web of Science database [53].

Soft humanoid robots are a new type of robot that combine the advantages of humanoid robots and soft robotics. For example, humanoid robots alone have a hard body, which limits the type of tasks they can perform, while soft robotics can adapt to different environments and tasks. Soft humanoid robots can bridge the gap between the two technologies, enabling robots that are both adaptive and capable of performing complex tasks. Potential applications for soft humanoid robots include search and rescue, medical and industrial applications, and human-robot interaction.

To create successful soft humanoid robots, researchers will need to develop methods to integrate existing technologies from both humanoid and soft robotics. This will involve addressing issues such as power and control, materials selection and safety. Research on soft humanoid robots will also require the development of new algorithms, such as those for motion

planning, control and machine learning.

In particular, for humanoid robots the trend is to investigate the introduction of soft robotics in a gradual way, resulting in hybrid robots.

On the one hand, it is possible to distinguish projects that emulate soft human parts that do not aim to replace their equivalent in a rigid humanoid, such as manipulator robots. In these cases, the development is not subject to the constraints of future integration into a humanoid. The project described in [54] at the University of Hong Kong consists of a manipulator. The BCL-13 robot, shown in Figure 2.17, is a four-fingered hand capable of grasping various objects with up to 13 degrees of freedom. Another project that is very different from a manipulator is that developed by the National University of Singapore [55], which propose the use of soft actuators in robotic eyes.

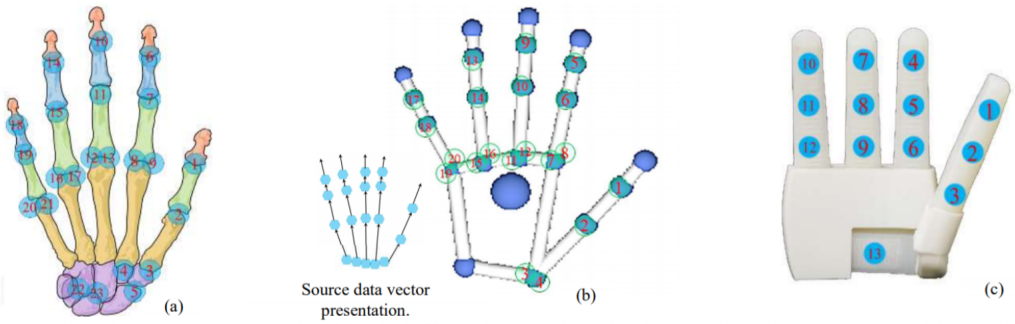


Figure 2.17: BGL-13 Robot Soft Hand [54].

However, the incorporation of these soft elements into existing rigid humanoid robots is also being investigated. In 2002, [56] published the integration and control of a spine design for the humanoid Kenta from the Department of Mechano-Informatics at the University of Tokyo. The spine is made up of links or vertebrae, which can bend by the action of tendons, behaving in a very similar way to a human spine. In total, it is composed of 10 joints, each with three degrees of freedom, which give the humanoid greater mobility and more human-like movements. Figure 2.18 shows the spine already integrated into the Kenta humanoid.

Rollin JUSTIN (DLR, Germany), a humanoid robot with up to 51 degrees of freedom, was initially designed as a starting point for research into complex manipulation tasks. It was primarily oriented towards the development of technology that could be used to create robots that could assist with domestic tasks. With the integration of Adaptive Whole Body Manipulation, the Justin robot is able to perform various tasks that mimic human manipulation [57]. A full body image of this humanoid is shown in Figure 2.19. A modular 3-DOF prototype equipped

Figure 2.18: Kenta: a whole-body tendon-driven flexible-spine humanoid [56].

with different tendon shapes and elastic continua is designed and built as shown in Figure 2.20. The performance of the prototype was evaluated on a planar bench [58]. The proposed design focuses on finding a suitable design for a robotic spine and providing a bench to develop control strategies for this mechanism [59], [60].

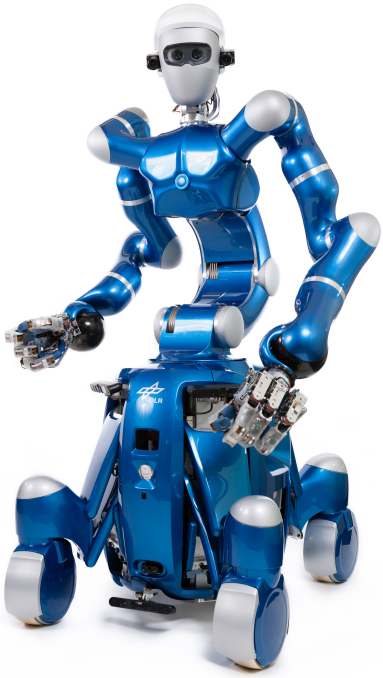


Figure 2.19: Humanoid robot Rollin' Justin developed by DLR.

TEO (Task Environment Operator) is a full-size humanoid robot developed since 2012 by the RoboticsLab research team at the University Carlos III of Madrid. It is highly optimized and energy efficient, taking advantage of several years of humanoid research, including its predecessor, Rh-1 robot [61]. TEO is therefore the result of a number of research projects, some

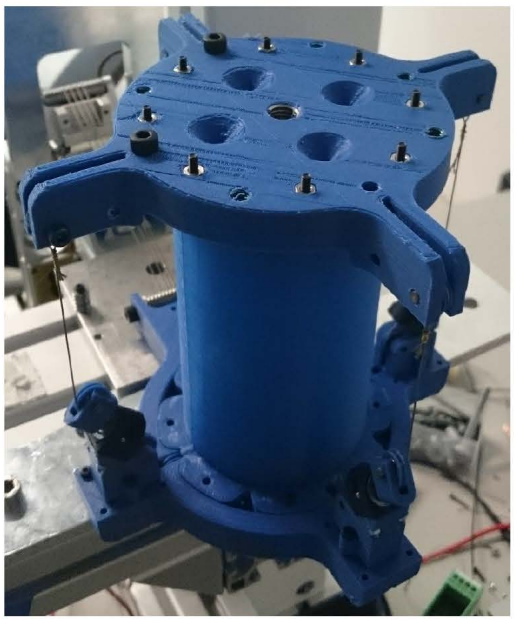


Figure 2.20: 3 DOF modular setup [58].

of which are ongoing, which means that the humanoid is constantly evolving. TEO is approximately 1.70 metres tall and weighs 60 kilograms, making it very similar in size to an adult human. It is designed to replicate some of the tasks that a human can perform [62]. Figure 2.21 shows the current status of TEO in the laboratory, where it has been also researched for the HumaSoft project.

2.4 HumaSoft project

This PhD. thesis is framed within the HumaSoft project, funded by the Ministry of Economics, Industry and Competitiveness, from 2016 to 2020. The thesis is totally aligned with the project's objective, which was to develop a soft link technology for use in humanoid robots. The project contributed to the objectives of the Horizon 2020 Call ICT-2016 Work Programme, mainly to those related to soft manipulation and safe interactions [63].

Particularly, this thesis addresses the design, modeling and development of a soft joint acting mainly as a robotic neck to be integrated into the humanoid robot TEO in order to increase its adaptability and flexibility and to provide a safer interaction with the user and the environment.

TEO's neck is currently rigid and has 2 DOF that supports the head of the robot, which consists of a Kinect camera housing that houses the PointGrey FLEA monocular camera and an ASUS XTion Pro Live RGB sensor.

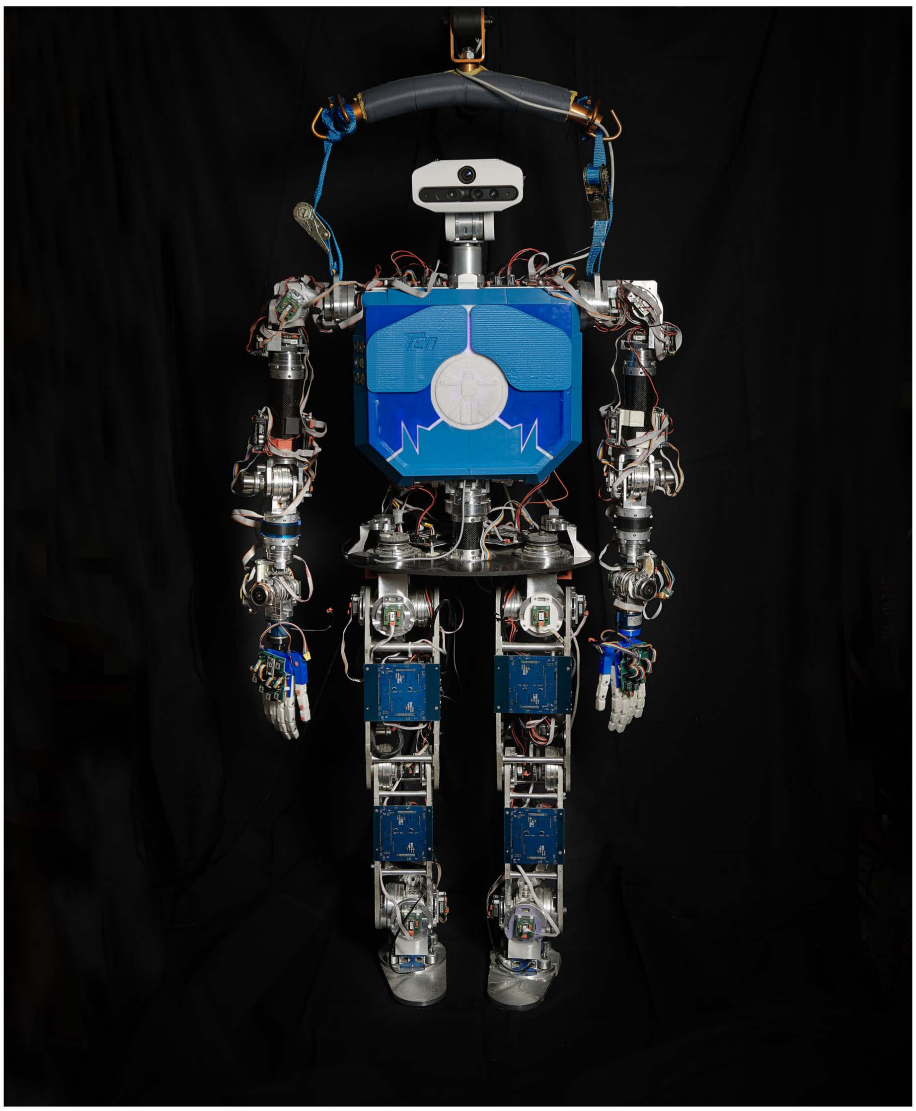


Figure 2.21: Humanoid robot TEO.

The goal of the following chapter is to introduce a first approach to the soft robotic neck design and to show a research path in this direction. The development of the neck platform is independent of TEO, but the technology developed is fully compatible and integrated into the humanoid, which is one of the principles of the HumaSoft research.

Chapter 3

DESIGN OF SOFT JOINTS

In this chapter a soft robotic cable-driven mechanism is proposed with the purpose of later creating softer humanoid robots that meet the characteristics of simplicity, accessibility and safety. It is the purpose for this soft link to be used interchangeably in various limbs of the humanoid robot, like arms and neck, under the constraints of scalability, controllability of their stiffness and integration. The first step towards this goal is the design and performance analysis of a prototype of soft link working as a neck, with definition of its material and its actuation system. A redesign will then be performed on the previously mentioned soft neck prototype, with the goal of its incorporation into the humanoid robot TEO.

Finally, a design for a soft robotic arm is presented, reusing the principles used in the design of the soft neck, focusing on the versatility of the soft link to be used in different parts of the humanoid robot.

3.1 Design requirements for the soft neck

The prototype is the result of research on robot necks, mainly in the configuration of parallel robots [64]. The motions of the human neck include flexion / extension, vertical rotation and lateral flexion [65], as shown in Figure 3.1. The flexion movement with a range of 50°, while the extension has a range of 57.5°, ensures that the head is tilted backwards. The vertical rotation of the neck is described as the rotation of the head to the right and to the left with a rotation range of 70°. The lateral flexion allows the head to bend towards the shoulder, with a maximum angle of 45°.

The purpose here is to design a soft robotic neck of two DOF (extension-flexion and lateral

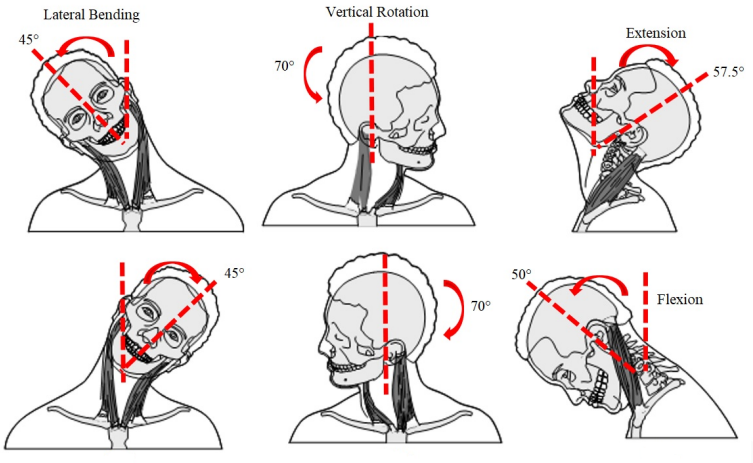


Figure 3.1: Human neck motions.

bending) with a maximum lateral bending of 40°, under a criterion of flexibility and inspired by the human neck structure, using a column spring. These two DOF will be later redefined as inclination and orientation when describing the mechanical movements of the neck platform. The mechanism will support a load of $1Kg$. The cable-driven continuum mechanism should be able to provide a range of motion similar to that of a human neck, while also providing a high torque output, allowing it to move the robotic neck with a large force.

The design of the prototype will follow the stages shown in Figure 3.2: selection of the cable-driven parallel mechanism, design of the mechanical system, requirements of the electrical system and design of the overall soft neck prototype. All these stages will be described in detail in the following sections.

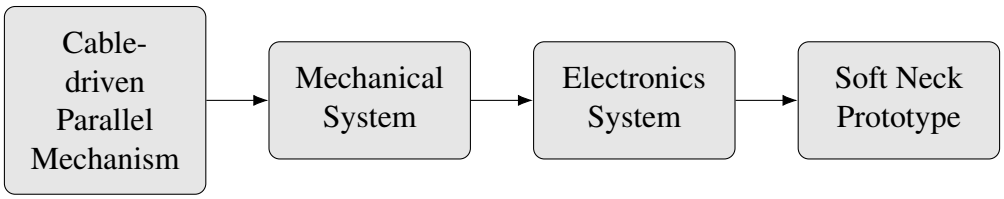


Figure 3.2: Diagram of the design steps.

3.2 Cable-Driven Parallel Mechanism (CDPM)

A manipulator or parallel mechanism in a generalized definition is a closed-loop kinematic mechanism where the moving element is linked to the base by several independent kinematic chains.

In the first design phase, an explanation of the elements that make up a cable-driven parallel

mechanism will be made, and the number of links needed to obtain the 2 DOF platform will be determined. In the Mechanical System stage, the dimensions of the spring to be used in the mechanism will be determined, and additionally the tension force of the cable will be calculated, which will allow sizing the electric actuator to be chosen. In the Electrical System stage, the electric motor will be dimensioned according to the parameters calculated in the previous stage. Finally, in the fourth stage, the CAD sketch of the robotic neck is made with the respective functional explanation of each element that makes up the prototype.

3.2.1 General structure of the CDPM

The general structure of these manipulators includes a base platform, a moving platform or end effector, and an extendable or telescoping central column that connects the base to the moving platform to apply a push force to the platforms. The central column can apply force by means of an actuator (active) or spring or pistons (passive). Robotic manipulators use a combination of active and passive tensioning elements (cables) and rigid links to maximize the benefits of both pure cable and conventional parallel mechanisms.

The general configuration of cable-based manipulators is studied in [64] and a general schematic is shown in Figure 3.3. The four main elements of these manipulators are:

- **Base:** the fixed part of the manipulator to which the global coordinate system $OXYZ$ belongs.
- **End-effector:** the moving body containing the coordinate system $O'X'Y'Z'$.
- **Cables:** flexible elements with negligible mass and diameter connected from one end to the end-effector at points P_i (i=1,2, ..., m) and pulled from the other end at Q_i. The pulling actuator produces tension t_i inside the cable and can be simply a winch that pulls and coils the cable or a separate mechanism that moves the end of the cable (Q_i) without changing its length. The unit vectors $\hat{u}_i$ (i=1,2, ..., m) determine the direction of the wires and point towards the base. Depending on the manipulator structure, there may be some additional sheaves to guide the cables. The number of cables m is equal to the dimension of the moving body motion space.
- **Spine:** element that produces a force between the base and the moving body to keep all cables in tension. The spine can be an active element that generates a desired force.

It can also be a passive element, such as a pneumatic cylinder or a compression spring suitably designed to provide sufficient force required to maintain tension on the cables. The direction of the backbone is shown by unit vector $\hat{w}$ pointing towards the end effector.

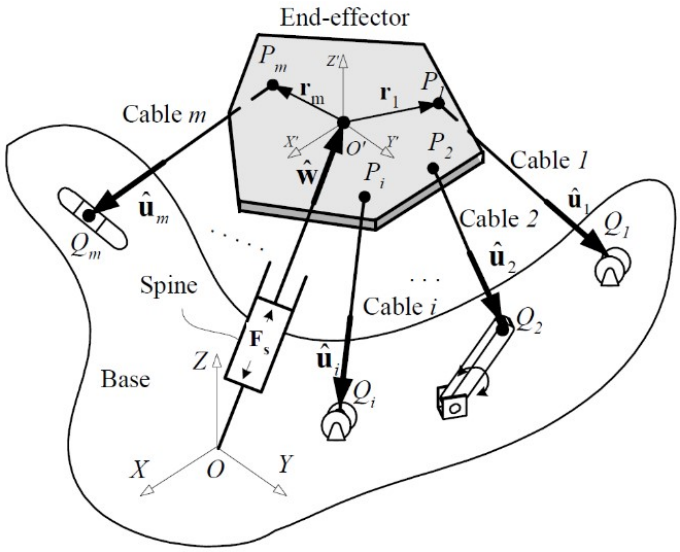


Figure 3.3: General configuration of a CDPM.

Parallel robots for which the number of chains is strictly equal to the number of DOF of the end-effector are called fully parallel manipulators. Gosselin [66] characterizes fully parallel manipulators by the equation

$$p(n-6) = -6 \tag{3.1}$$

where p represents the number of DOF and n the number of rigid bodies within a chain. The objective here is to reach the two DOF of the prototype according to the design specifications defined above. Therefore, substituting $p = 2$ in Equation (3.1), $n = 3$ is obtained; that is, a three cable-driven system is needed to develop the two DOF parallel mechanism.

3.3 Spring-based neck

In section 3.2.1 it was determined that for the parallel mechanism to be develop in this work three cables are needed. Figure 3.4 shows a general outline of the prototype to be designed.

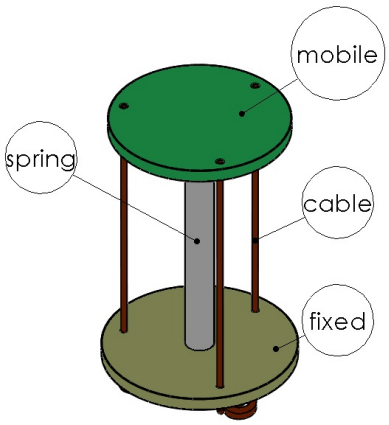


Figure 3.4: Sketch of the three cable parallel mechanism.

3.3.1 Mechanical system

A spring will be used as a central column, which must be suitably designed to provide the sufficient force required to maintain the tension in the cables and to buckle around the neutral axis to generate a two DOF rotation. The material of the spring is ASTM A228 and the equation that describes the deflection of the spring can be approximated to

$$\delta = \frac{8FD^3N_a}{Gd^4} \tag{3.2}$$

where δ is the compression deformation; D is the diameter of the spring (helix); d is the diameter of the wire; Na is the number of coils; G is the shearing modulus; E is the elastic modulus and F is the applied load of $1Kg$ as a design requirement. Table 3.1 presents the values of the parameters of the selected compressive spring.

Table 3.1: Parameters of the selected compressive spring.

$\delta(m)$	L_o(m)	$d(m)$	Na	$F(N)$	$G(GPa)$	$E(GPa)$
0,001	0.1	0.003	15	9,8	80	200

3.3.1.1 Cable pulling force

Once the spring has been designed, it is necessary to know the force required for the mechanism to reach the required angle of inclination. The maximum force will be provided when the mechanism gets 40° of inclination, which was previously defined as the maximum bending angle of the spring.

The force study will be based on the static analysis of the mechanism with one DOF [67],

inspired by the bending motion of the human neck developing an analysis on the lateral buckling of the robotic neck. The inclination/orientation motion mechanism of the humanoid neck is a practical compressed helical spring lateral buckling problem, as discussed in the work by Timoshenko [68], who investigated the compression, bending and shear stiffness coefficients of a helical spring to explain the stability limits. Furthermore, Timoshenko also pointed out that the lateral buckling of a coil compressed spring can be studied by the same methods as prismatic bars, but considering the change in spring length due to compression. The following notations of the coil spring are:

L_0 is the initial spring length;

N_a is the number of coils;

d is the wire radius;

L is the spring length after compression;

I is the moment of inertia of the cable cross-section with respect to its diameter;

$\alpha_0, \beta_0, \gamma_0$ are the compression, bending and shear stiffness coefficients of the non-loaded helical spring; and

$\alpha_s, \beta_s, \gamma_s$ are the stiffness coefficients of the loaded helical spring.

Timoshenko [68] mentions that the number of coils by unit length of the spring increases with the factor Lo/L when the spring is compressed, and the stiffness decreases by the factor L/Lo. Therefore, the following equations are reached:

$$\alpha_s = \alpha_0 \frac{L}{L_0}, \qquad \beta_s = \beta_0 \frac{L}{L_0}, \qquad \gamma_s = \gamma_0 \frac{L}{L_0} \tag{3.3}$$

The compression, bending and shear stiffness without load can be calculated by Equation (3.4), respectively.

$$\alpha_0 = \frac{GIL_0}{\pi N_a \left(\frac{d}{2}\right)^3}, \qquad \beta_o = \frac{2EGIL_0}{\pi N_a \frac{d}{2}(E+2G)}, \qquad \gamma_o = \frac{EIL_0}{\pi N_a \left(\frac{d}{2}\right)^3} \tag{3.4}$$

The lateral deflection of a spring is presented in Figure (3.5). The total load from the center of the moving base is considered as a point mass m_0, the mechanism is driven by two wires applying tensions T_1 and T_3 located from a distance a measured from the center of the moving base, and the wires pass through the fixed plate at a distance b measured from the center of the base.

As previously mentioned, the spring will be studied as a prismatic beam, the buckling

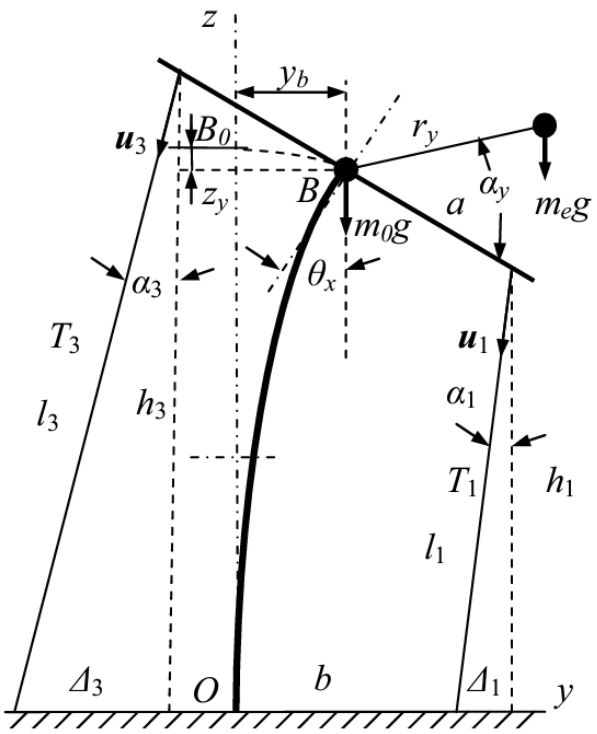


Figure 3.5: Lateral bending model of the CDMP [67].

moment in any part of the spring cross section is M and, by mathematical developments, it is known that the deflection curvature is given by

$$M = \beta_s \frac{\frac{d^2y}{dz^2}}{\left(1 + \left(\frac{dy}{dz}\right)^2\right)^{3/2}} \tag{3.5}$$

The total moment M applied in the bending of the spring consists of four parts:

$$M = M_{mo} + M_{me} + M_{T1} + M_{T3} \tag{3.6}$$

where:

- M_{mo}: Torque generated by the fixed base mass.
- M_{me}: Torque generated by external payload.
- M_{T1} y M_{T3}: Torque generated by cable tensions T_1 and T_3.

For small deflections Equation (3.5) can be written as a linear case:

$$M = \beta_s \frac{d^2y}{dz^2} \tag{3.7}$$

The adult human head can tilt more than 40º in deflection, the head motion generally in our daily life is no more than 15º [69]. Therefore, it is feasible to use the linear equation (3.7) to analyze the static model of the CDPM mechanism. Considering the initial conditions

$$(y)_{z=0} = 0, \qquad \left(\frac{dy}{dz}\right)_{z=0} = 0$$

the general solution of the equation (3.7) is as follows:

$$y = (y_b + a_0)(1 - \cos wz) - \frac{a_1}{w}\sin wz + a_1 z \tag{3.8}$$

where:

$$w = \sqrt{\frac{m_0 g + m_e g + T_1 \cos\alpha_1 + T_3 \cos\alpha_3}{\beta_s}} \tag{3.9}$$

$$\begin{aligned} a_0 = \big[& m_e g r_y \cos\alpha_y + T_1 \cos\alpha_1 a \cos\theta_x + T_1 \sin\alpha_1 a \sin\theta_x \\ & - T_3 \cos\alpha_3 a \cos\theta_x - T_3 \sin\alpha_3 a \sin\theta_x - (T_1 \sin\alpha_1 + T_3 \sin\alpha_3) L \big] \\ & (m_0 g + m_e g + T_1 \cos\alpha_1 + T_3 \cos\alpha_3)^{-1} \end{aligned} \tag{3.10}$$

$$a_1 = (T_1 \sin\alpha_1 + T_3 \sin\alpha_3)(m_0 g + m_e g + T_1 \cos\alpha_1 + T_3 \cos\alpha_3)^{-1} \tag{3.11}$$

The following equations describe the deflection angle of the moving base and the spring as a whole, as well as mathematically describe the different variables shown in Figure 3.5.

$$y_b = \frac{1 - \cos wL}{\cos wL} a_0 - \frac{a_1}{w}\tan wL + \frac{a_1 L}{\cos wL} \tag{3.12}$$

$$\theta_x \approx \left(\frac{dy}{dz}\right)_{z=L} = a_0 w \tan wL + a_1 (1 - \sec wL + wL \tan wL) \tag{3.13}$$

$$\begin{aligned} z_y = & \frac{\sec^2(wL)}{8w} \Big\{ -4a_0 a_1 w + 8a_1 (a_0 + a_1 L) w \cos wL + 2(a_0 + a_1 L)^2 w^3 L \\ & - 4a_1 L (a_0 + a_1 L) w^2 \sin wL - 2a_1 (2a_0 + a_1 L) w \cos(2wL) \\ & - \left[3a_1^2 + (a_0 + a_1 L)^2 w^2 \right] \sin(2wL) \Big\} \end{aligned} \tag{3.14}$$

$$\alpha_1 = \arctan\frac{\Delta_1}{h_1} \tag{3.15}$$

$$L_1 = \sqrt{\Delta_1^2 + h_1^2} \tag{3.16}$$

$$\alpha_3 = \arctan \frac{\Delta_3}{h_3} \tag{3.17}$$

$$L_3 = \sqrt{\Delta_3^2 + h_3^2} \tag{3.18}$$

The length L of the compressed spring is calculated as

$$L = L_0 - \Delta L \tag{3.19}$$

$$\Delta L = \frac{1}{K}\left[m_0 g + m_e g + \sum_{i=1}^{2} T_i \cos \alpha_i\right] \tag{3.20}$$

where K is the spring elastic constant. Parameters L_0, K, a, b, m_0, m_e and (r_y, α_y) are known, and eight independent equations from (3.12) to (3.19) are given, where θ_x is the bending angle; L is the compressed spring length; L_1 and L_3 are the cable lengths L1 and L3; z_y is the vertical displacement of the spring under buckling; y_b is the horizontal displacement of the spring under buckling; and T_1 and T_3 are the cable tensions.

However, there is a redundancy problem because, for a particular buckling angle θ_x, nine unknown variables must be found. Focusing on the tensions of the cables T_1 and T_3, a simple solution will be used, which consists of using only one cable to perform the buckling of the spring. Therefore, as shown in Figure 3.5, tension T_1 is sufficient for the lateral buckling, and tension T_3 will be zero. Furthermore, for this analysis the external load m_e will not be considered, consequently $m_e = T_3 = 0$, and the remaining unknown variables can be calculated.

Besides, the inertia I, the spring elasticity constant K and the bending constant β_0 are calculated as follows:

$$I = \frac{\pi d^4}{64} \tag{3.21}$$

$$K = \frac{Gd^4}{64 N_a \left(\frac{D}{2}\right)^3} \tag{3.22}$$

These additional spring constants are presented in Table 3.2, with $a = b = 0.05m$, $me = T_3 = 0$.

Table 3.2: Spring constants.

$I\left(m^4\right)$	$K\,(N/m)$	β_0
3,9761x10^{-12}	1.9825x10^3	0.0505

Using the ©Matlab Optimal Tool with the Solver function [70], the system of Equations (3.12) to (3.19) can be solved, considering θ_x as the input variable and the cable tension T_1 as the output variable.

The equation obtained for the linear fitted curve relating the bending angle θ_x and the cable tension T_1 is

$$T_1 = 1.338\theta_x - 4.872 \tag{3.23}$$

For a maximum angle of inclination $\theta_x = 40°$, as specified in the design, the corresponding cable tension is $T_1 = 60,392N$. This tension will be used in Section 3.3.2 to dimension the motors required to control the movement of the CDPM.

3.3.2 Electronic system

The motors to be used in the robotic neck will be DC motors, which allow a proper position and velocity control. Controlling the movement of the CDPM mechanism with two DOF requires three cables (Section 3.1.2) connected to the fixed and the mobile bases. Each cable will be connected to a drive unit including motor, gearbox and encoder that will feed the position back to the controller.

In [71] the different transmissions that a DC motor can perform are reviewed. The mechanical transmission from angular to linear is the most appropriate for our solution because the motors will have the function of pulling or releasing the cable to perform the lateral bending movement of the spring. While [71] explains three different forms of angular to linear transmission, the crane transmission is chosen because it fits our design requirements, as shown in Figure 3.6.

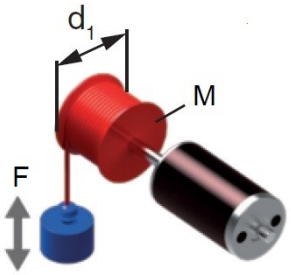


Figure 3.6: Crane transmission.

The load attached to the cable (blue element in Figure 3.6) represents the load force F_L necessary to bend the neck through a rotating coupling of diameter $d_1 = 20mm$ that will be

in charge of driving the cable of the mechanism. Finally, the torque M necessary to apply the maximum load force corresponding to an inclination of $\theta_x = 40°$ can be calculated using Equation (3.24), with an efficiency of 85%, taking into account the existence of friction between the components.

$$M = \frac{d_1}{2}\frac{F_L}{n} \tag{3.24}$$

where M is the torque, F_L is the load force, d_1 is the coupling diameter and n is the efficiency. Table 3.3 presents these parameters for the crane transmission.

Table 3.3: Parameters of the crane transmission.

$F_L\,(N)$	$d\,(mm)$	$n(\%)$
60,392	20	85

Solving Equation 3.24 a torque $M = 0.71Nm$ is obtained. With the help of the Maxon selection program [72], the selected motor is RE 273757 with Planetary Gearhead 166155. For the control of the DC motor, the driver IPOS 4808 MX will be used through CAN communication.

3.3.3 Parallel prototype

The designed CDPM mechanism is shown in Figure 3.7, which represents the proposed prototype of the soft robotic neck of two DOF.

Table 3.4: Elements of the soft robotic neck.

	Element
1	Mobile Platform
2	Fixed Base
3	Spring
4	Pulley
5	Cable
6	Rotating Coupling
7	Motor Base
8	Planetary Gearhead
9	DC Motor
10	Platform
11	Mounting Base

Table 3.4 describes the different parts of the system and Table 3.5 presents the general specifications of the prototype.

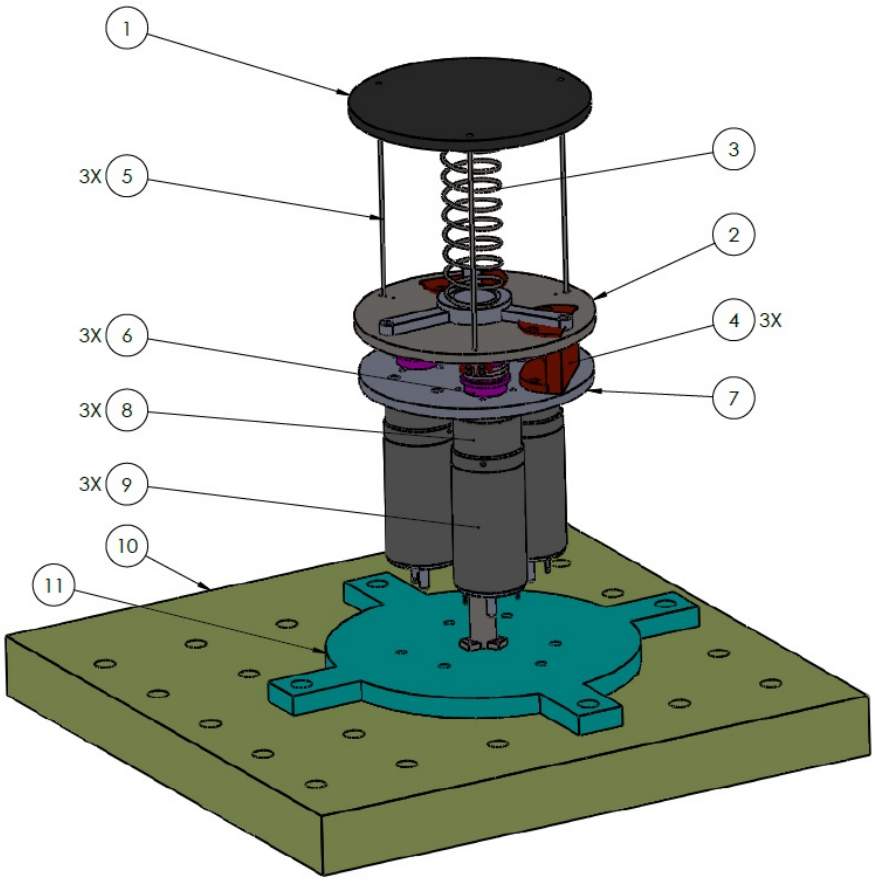


Figure 3.7: Prototype functional scheme.

A compressive spring serves as the main mechanical structure of the soft robotic neck. Element 2 acts a a fixed base, while Element 1 is a mobile platform driven by three symmetrically distributed cables that are pulled by the DC motors located below the fixed base.

Table 3.5: General specifications of the neck prototype.

Technical specification	Value
Load capacity	1Kg
Degrees of freedom	2
Voltage	48V DC
Approximate weight	1.9Kg
Type of mechanism	CDPM
Approximate dimensions	0.11 x 0.11 x 0.262m
Tilt range	0°- 40°
Sensors	Encoders
Actuator	DC Motor: Maxon RE 273757

A winch will be used to guide the cables, whose configuration is shown in Figure 3.8. The Maxon RE 273762 servomotor integrated with an absolute encoder is coupled to a planetary

gearbox with a gear ratio of 3.7:1. This drive train is connected to a rotating coupling with diameter $d = 20m$ that can storage a cable length of up to $60mm$. As a result of the equal pitch of the rotating coupling and the spindle, the relative direction of the spiral cable is constant, which allows a reliable winding and unwinding of the cable. This is especially important because the cable speeds and accelerations will be high.

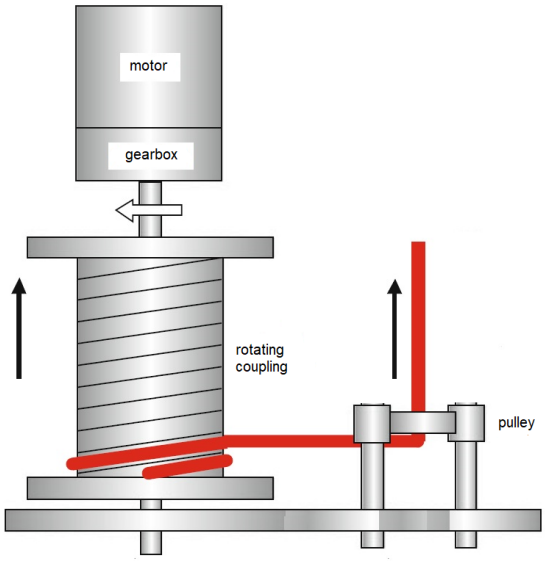


Figure 3.8: Winch configuration.

3.3.4 Design of mechanical elements

Now the different mechanical elements constituting the neck prototype are described, each of which plays an important role in the correct performance of the robotic neck.

3.3.4.1 Mounting base

This part of the mechanism is designed to keep the actuators fixed and also to support the whole system. The resulting rigid base with four outlets to be fixed to the existing platform is designed as shown in Figure 3.9.

3.3.4.2 Mobile platform

The most important element of the structure is the mobile base as this will allow the simulation of the neck movements. For the dimensions of the base, it will be taken into account that the actuators will be distributed at 120° each, so a circular base with a diameter of $10cm$ will be designed, as shown in Figure 3.10.

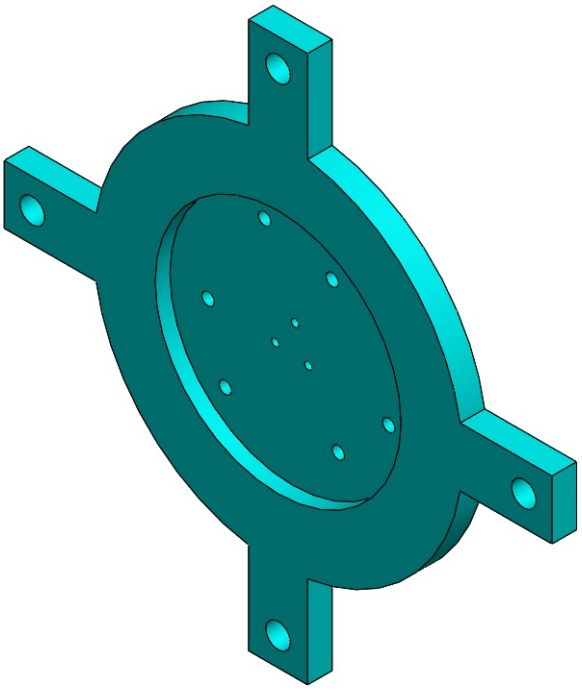


Figure 3.9: Mounting base.

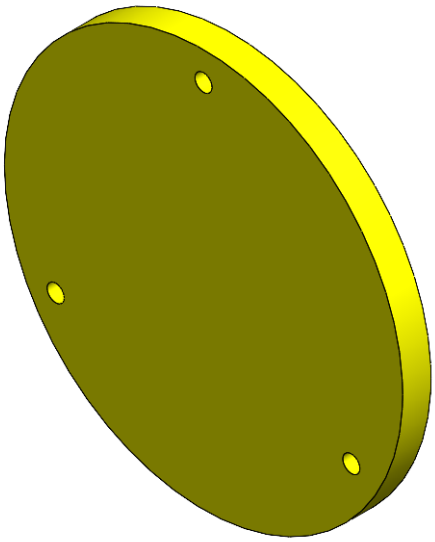


Figure 3.10: Mobile platform.

3.3.4.3 Fixed base

The function of the fixed base is to hold the fixed end of the spring, and it will have a column to attach to the mounting base. It will have the same dimensions as the mobile base, as shown in Figure 3.11.

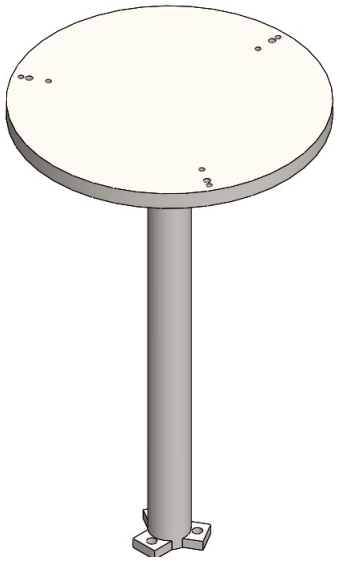


Figure 3.11: Fixed base.

3.3.4.4 Pulley

As mentioned above, the pulley is designed to guide the cable that is winding or unwinding on the rotating coupling, see Figure 3.12.

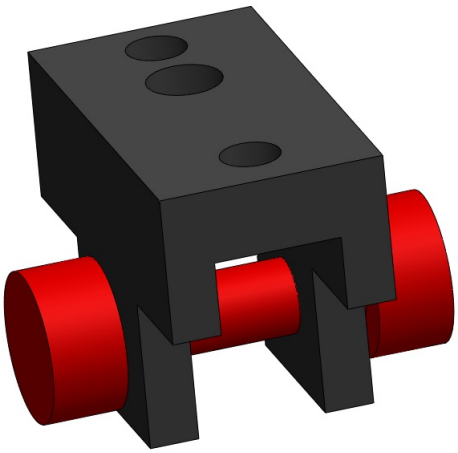


Figure 3.12: Pulley.

3.3.4.5 The rotating coupling

Finally, this part will be coupled to the spindle of the gearbox and has the function of winding and unwinding the cable, see Figure 3.13.

The manufacturing material for the mechanical components is polylactic acid (PLA), which is commonly used in 3D printing. PLA is a biodegradable polymer derived from lactic acid and its production is 100% renewable resources.

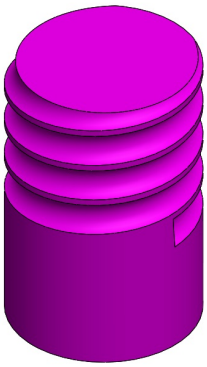


Figure 3.13: The rotating coupling.

The resulting prototype is the one presented in Figure 3.14.

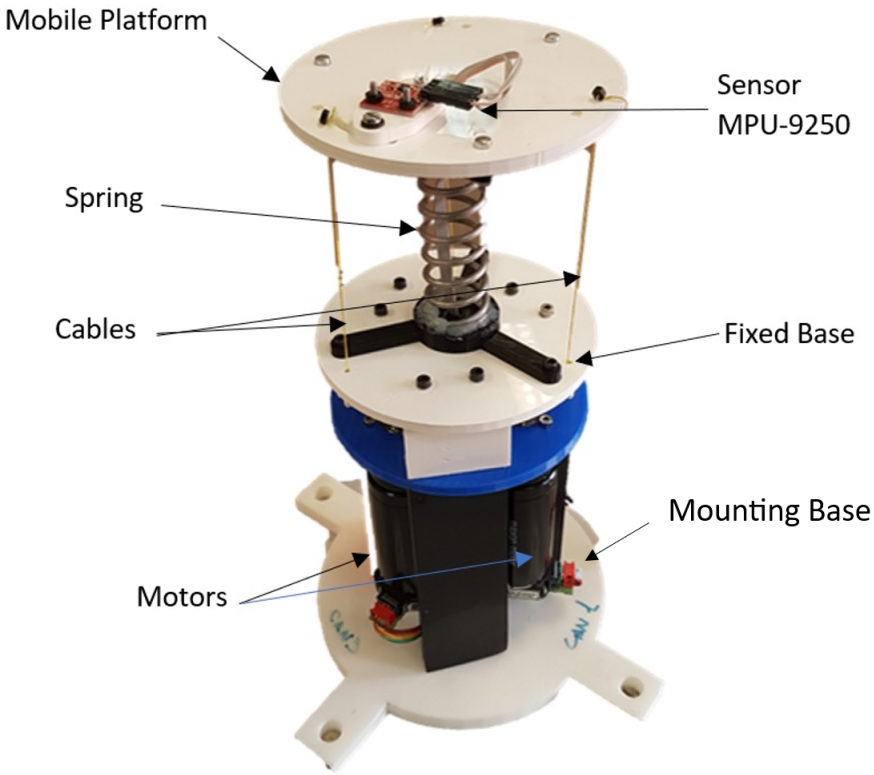


Figure 3.14: First physical model built.

3.3.5 Hardware connection of the platform

Once the platform has been assembled, the hardware connection has to be addressed, as depicted in Figure 3.15. The elements necessary for this connection are:

- **Platform.** Assembled with the three motors and their respective encoders and gearboxes.

- **Three iPOS4808 MX drivers.** Connected to the connection board that will allow the connection of the driver with the rest of the elements. Two iPOS can be connected to each board.

- **Two connection boards.** Both boards are used to connect on one side the signal coming from the encoder and the power supply of each motor, and on the other side it receives the power source and the CANbus signal.

- **Power source.** Providing a voltage that must be higher than 30V and lower than 48V, it supplies power to the complete system starting from the connection boards.

- **PeakCAN PCAN-minipci board.** Located inside the computer case, it allows the connection of the computer to the CAN network.

- **CPU.** Sends the CAN messages for each driver to regulate the movement of its engine.

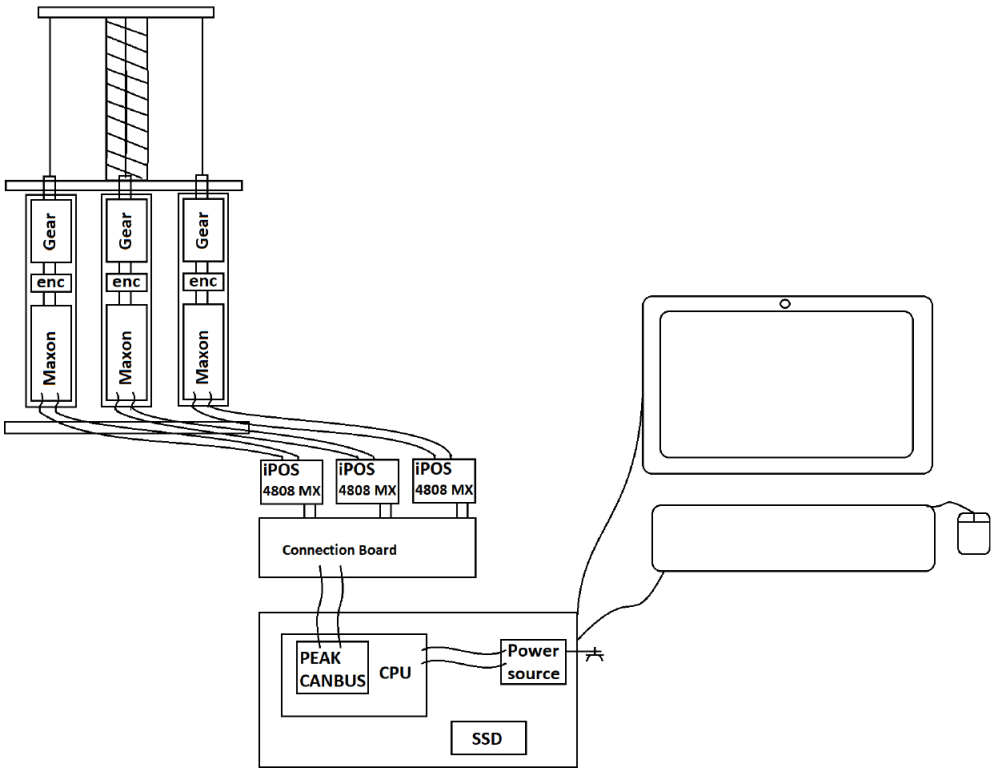


Figure 3.15: Hardware connection diagram of the platform.

3.4 Soft link-based neck

In an original concept, the soft link consisted of a spring because its deformation during bending is continuous and constant. However, the spring is not fully compatible with the objectives of HumaSoft because, although it deflects sufficiently, it is not made of a soft material and therefore could not provide safer and softer interactions as specified in [63]. For this reason, HumaSoft has focused its research on real soft links, with bio-inspired vertebral geometry.

In the following sections the replacement of the spring by the soft link is addressed.

3.4.1 Mechanical system

For the mechanical system the goal is to replace the spring used in the previous design by a soft link maintaining the dimensions of the spring. With regard to the manufacturing process of the link, a flexible material such as thermoplastic elastomer (TPE) or NinjaFlex (commercial name) will be used.

3.4.2 Soft link for the neck

In general, in order to maintain the same dimensions as the prototype, it was decided to design a soft link with the same dimensions as the spring designed in Table 3.1. For the initial design, a simple link has been considered as shown in Figure 3.16.

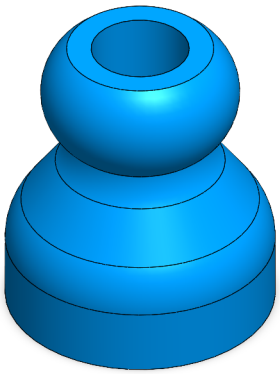
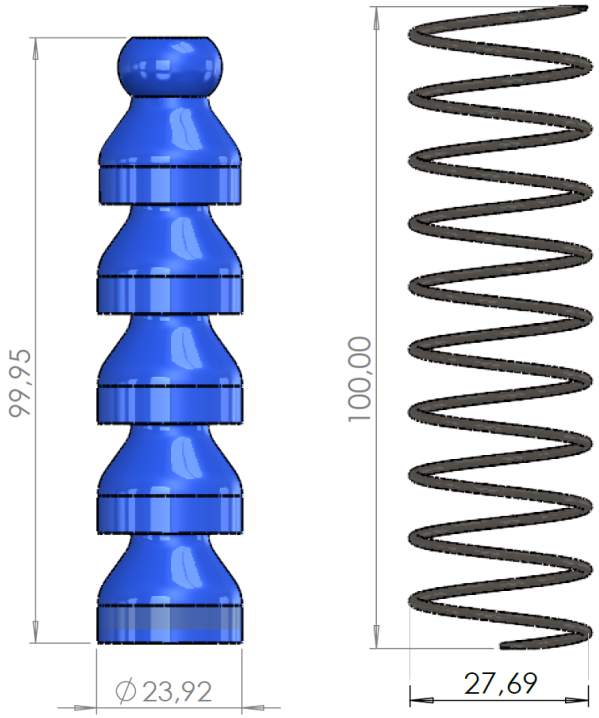


Figure 3.16: Simple link.

Once the simple link has been designed, five simple links will need to be joined together to form the main soft link. This will be done by 3D printing using Ninjaflex filament which will be bonded to the top of the single link to create a strong 10cm high complete link with the same spring height (Figure 3.17).



(a) Soft link. (b) Spring link.

Figure 3.17: Kinds of links [*mm*].

A soft link made of Ninjaflex can be an excellent replacement for a traditional spring. Ninjaflex is a flexible thread that has the same properties as a spring (Table 3.6) but with more flexibility.

Table 3.6: Mechanical properties of NinjaFlex®[73].

Property	Value
Young modulus (MPa)	12
Tensile Strength, Ultimate (MPa)	26
Yield strength (MPa)	4
Mass density ($Kg/m3$)	1040

3.4.2.1 Validation and simulation

To validate the soft link, a tension study is carried out to check that the strength applied to achieve the maximum flexion angle of 40º does not exceed the elastic limit of the link. The yield strength of Ninjaflex is 4Mpa. Figure 3.18 shows how the soft link, acting as a central column, reaches a flexion of 40º and is able to recover its original state, since the elastic limit does not exceed the theoretical one.

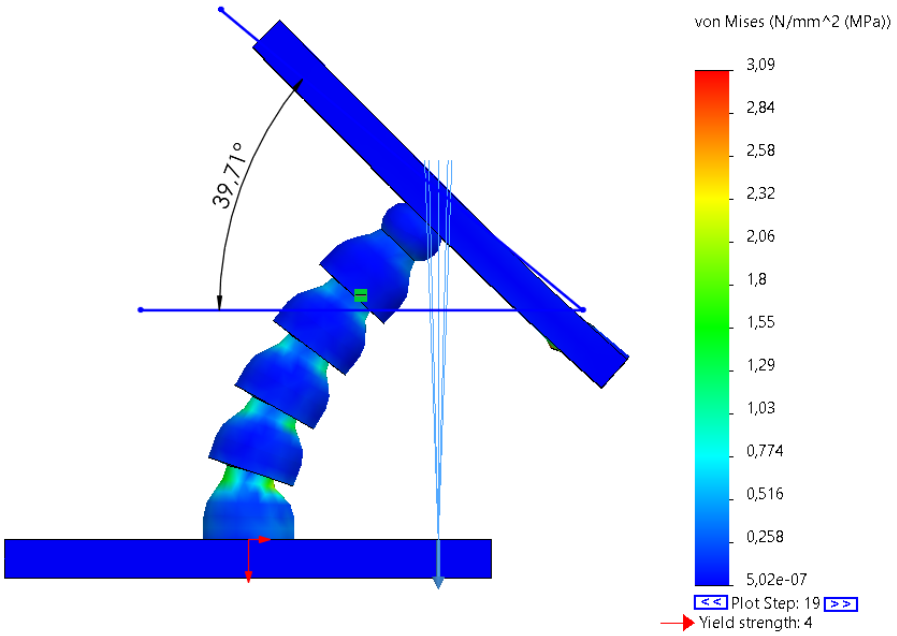


Figure 3.18: Tension study for soft link validation.

As mentioned at the beginning of the mechanical design of the soft link, the goal is to keep the mechanical design proposed in Section 3.3.3 unchanged, maintaining its flexibility and mobility. The new CDPM is shown in Figure 3.19.

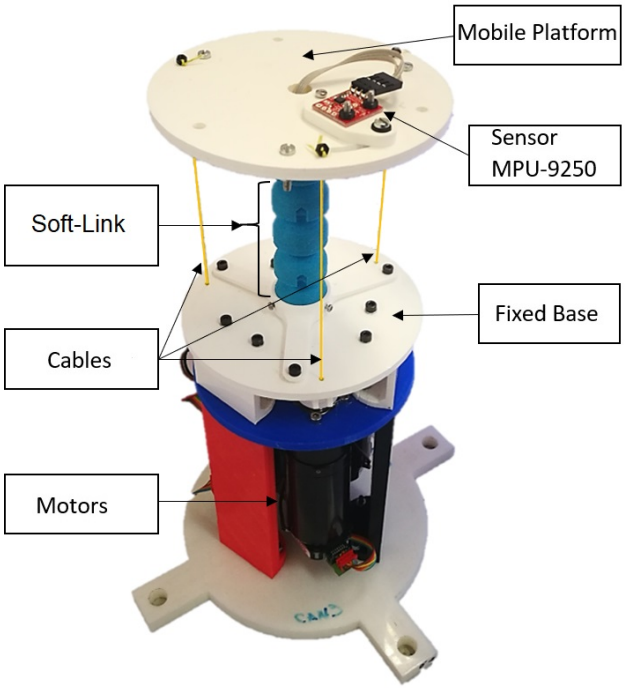


Figure 3.19: Soft link-based robotic neck.

Now the goal is to integrate this soft link-based robotic neck into the humanoid robot

TEO. For this integration the redesign of the prototype is needed. In the following section, the mechanical and electronic redesign of the neck will be performed. This will improve the functionality and security of the device and allow a better coupling between the humanoid robot and the soft neck.

3.5 Implementation of the soft neck in the humanoid robot TEO

In order to integrate the neck into the robot, it is necessary to design a strategy for the combination of the hardware elements that compose the two robotic structures. Due to its nature, the strategy used to implement the mechanical connection is different from the one used to integrate the electronic components of the neck, and they will be described as mechanical integration and electronic integration, respectively.

Figure 3.20 shows a schematic of the integration of the neck into the humanoid. This integration maintains the DOF of the humanoid robot and its level of actuation, but gives it the advantages of soft robotics in terms of safety and acceptability.

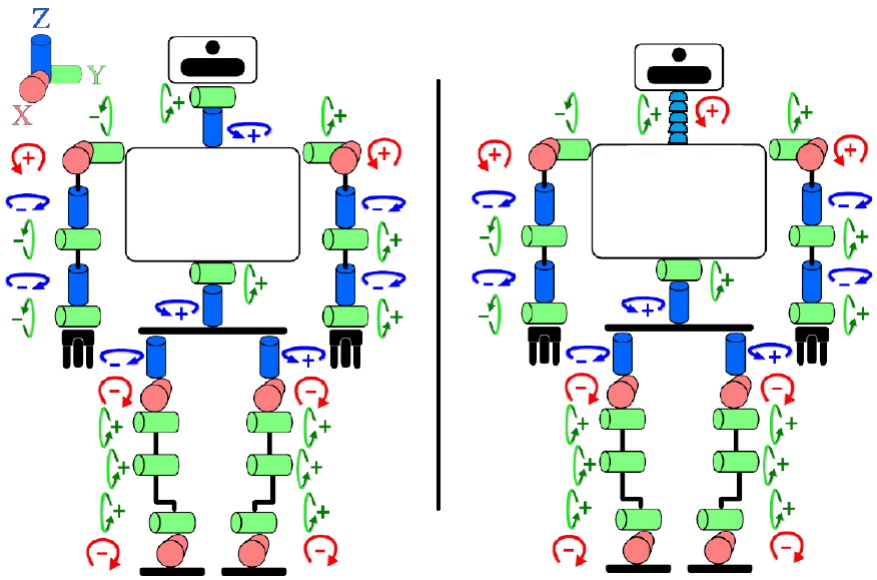


Figure 3.20: Schematic of the soft neck integration into the humanoid robot TEO.

3.5.1 Mechanical integration

This section describes the strategy for integrating the mechanical attachment of the neck into the humanoid. For this purpose, the different elements that physically compose the platform are presented, including the current mechanical attachment. The proposal for the mechanical integration and the final result are then presented.

Figure 3.21 shows the proposed redesign, with a soft neck of increased flexibility and several differences from the initial model (Figure 3.19). These differences include a change of motors to smaller ones, while maintaining their performance. This allows the robot to perform movements in confined areas without compromising efficiency.

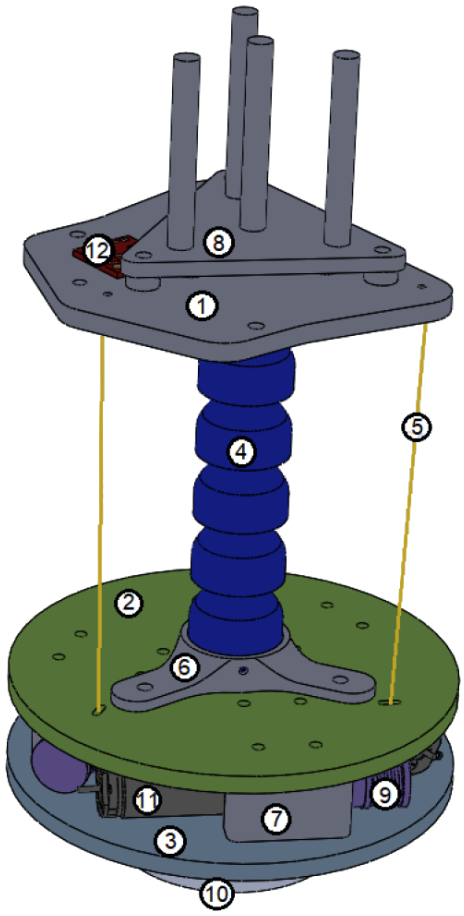


Figure 3.21: Final soft link-based robotic neck.

3.5.1.1 Detailed description of the platform elements

The elements that compose the platform are described below with reference to Figure 3.21:

1. **Mobile base:** This is the moving plate of the platform, which inclines in a certain orien-

tation when a position is commanded.

2. **Neck base**: This is a fixed plate of the platform on which the soft link is positioned. This part defines the space between the motors and the moving part of the neck.

3. **Motor base:** This plate supports the motors and other components for the correct mechanical operation of the platform.

4. **Soft link:** This link, made by 3D printing in Ninjaflex polymer, is bio-inspired by vertebrae to allow a continuous and uniform bending. The platform can operate with different soft links depending on their geometry, material and density.

5. **Cables:** The cables must be thin, strong and inelastic. For this reason, a wire similar to a fishing line was used as a reference.

6. **Adapter:** This part allows easy exchange between soft links. Two adapters are required for each different neck. The adapters are fixed to the link with adhesive and, to exchange links, it is only necessary to screw the adapters of the new link to the platform.

7. **The lift:** This part has a double function: it fixes the motors in their operating position and also serves as a support between the base of the neck and the base of the motors.

8. **Mass support:** This structure allows small weights to be inserted to increase the mass supported by the platform during operation.

9. **Pulley:** Located on the axes of the motors, they store the cable excess to allow it to be collected or released. There is one pulley for each of the three motors.

10. **Interface:** This allows the mobile base to be screwed to a surface.

11. **Motor, Encoder and Gearbox:**

    - **Motor model Maxon RE 16-16 mm, Grato brushes, 4.5 watts:** One for each cable, these are the actuators of the system.
    - **Planetary gearbox GP 16 A -16 mm, 0.1-0.3 Nm 24:1:** It transmits power from the motor to the shaft, reducing the speed and thus increasing the force.
    - **MR type M encoder, 512 ppv, 2 channels, with line driver:** It reads the position of the motor shaft to ensure it is at the correct position or speed.

12. **Sensor:** It is an Inertial Measurement Unit, model Sparkfun IMU-MPU-9250, which allows obtaining data of the position of the platform in real time, which makes it possible to control the platform in closed loop.

3.5.1.2 Mechanical interface for the attachment of the soft neck to the humanoid robot TEO

In the initial version of the platform, described in Section 3.3.3, the base part is designed to be attached by screws to a support. However, this configuration is not compatible with the current robot design. Since TEO already has a rigid neck to support its current head, it is more appropriate to adapt the soft neck design and retain the neck design of the humanoid and, in this way, the exchange between the soft prototype and the old rigid neck is allowed. The idea of the use of adapters that was applied for the exchange of different soft necks on the platform itself is reused, introducing an element that acts as an interface between TEO and the different necks. The objective is to design, manufacture and evaluate a mechanical interface to fix the neck in a simple, compact and reversible way.

Previously to this work, a design was proposed for an interface that would allow easy removal of a neck and exchange it for another one. This interface consists of two parts mechanically joined together in a thread-like fashion, each part attached to the neck and to the attachment surface (torso), respectively.

However, this design did not have the necessary configuration to be screwed into TEO's torso. This required a redesign in which the screw holes on the bottom part of the adapter coincided with those of the humanoid. Figure 3.22 shows the rigid neck that is currently installed on the humanoid and the configuration of the screws that the interface must have to achieve the integration of the platform in the simplest way.

Finally, after making the above modifications to the initial interface design, the CAD drawings shown in Figure 3.23 and Figure 3.24 were obtained.

It can be seen that Figure 3.23 corresponds to the part that will be screwed to the neck, hence the configuration of the screws in three equidistant axes, at 120º with respect to the centre, following the line of many of the elements of the platform. On the other hand, Figure 3.24 corresponds to the bottom part of the interface, whose holes allow the platform to be screwed into the humanoid, without making any modifications to the design of the latter.

The parts were manufactured by the same method used for the majority of the platform

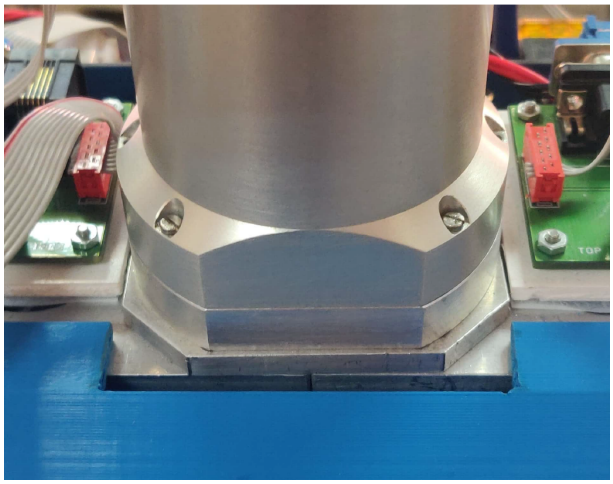


Figure 3.22: Current TEO rigid neck with the necessary screw configuration for the integration of the soft neck.

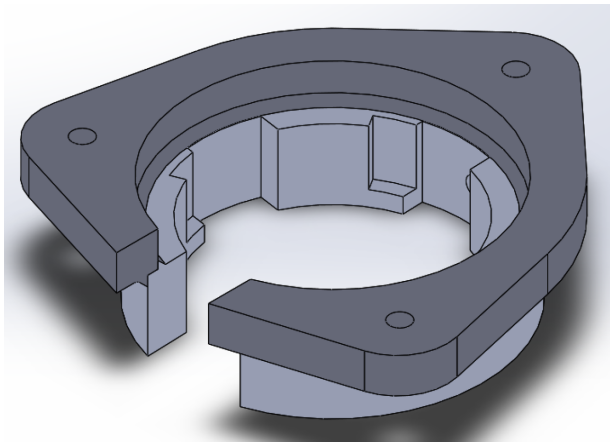


Figure 3.23: Design of the top part of the mechanical interface for the neck attachment.

elements: 3D printing using polylactic acid (PLA) filament. Figures 3.25 and 3.26 show the complete new interface, threaded and unthreaded, which allows the mechanical integration of the soft neck into TEO.

3.5.2 Electronic integration

The second part of the hardware integration is the integration of the electronic components into the soft neck, which are originally located outside the neck. The following sections address the design of the new electronic board.

3.5.2.1 Electronic connection redesign

The electronic connections present a number of difficulties in terms of their integration into TEO. Firstly, the large size of the connection boards needed to connect the drives has to be

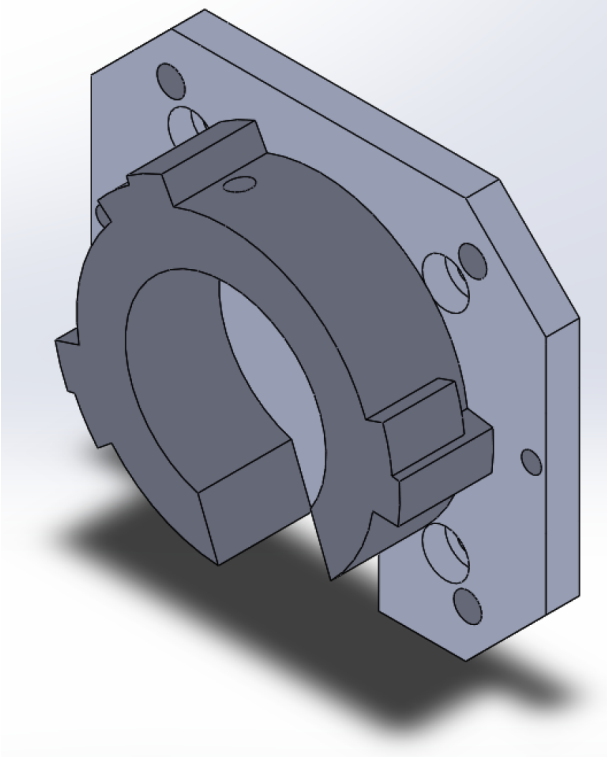


Figure 3.24: Design of the bottom part of the mechanical interface for the neck attachment.

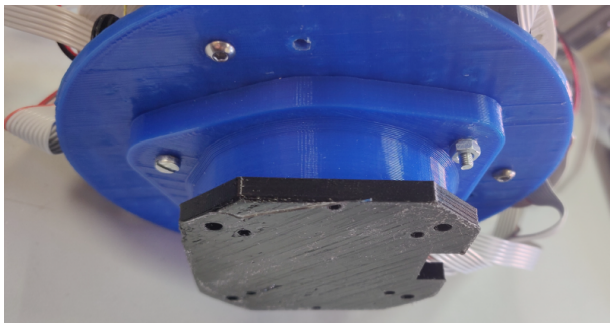


Figure 3.25: New interface parts installed on the soft neck in a closed state.

faced. In addition, the electronic connection is composed of numerous elements with several overhead cables. These two factors make the current model uncompact, making it very difficult to integrate the neck into TEO, where the space is limited.

For these reasons, a new design of the electronic connections of the platform is proposed. The first step is to follow the electronic connection diagram of the prototype, shown in Figure 3.15. It illustrates in broad terms which elements the electronic connections are to include and how these connections are to be made. However, the geometry, model and arrangement of these elements could be modified from the initial design.

The requirements for the new design are:

- **Compactibility:** As mentioned above, the new connection must be integrated into the

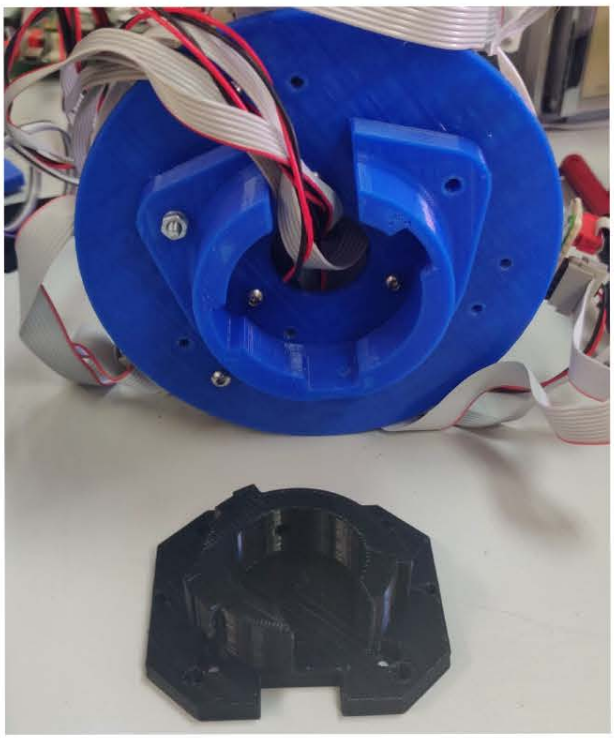


Figure 3.26: New interface parts installed on the soft neck in an open state.

platform and be small enough not to be bulky when implemented in TEO.

- **Compatibility:** The new connection design must be able to be used with iPOS drives, with CANopen communication, so that most of the software development of the neck can be retained, and at the same time can be integrated into the humanoid. It must also be considered that the platform will be powered by TEO's power lines, and that the drives will be additional nodes in the humanoid's CAN network.

- **Efficiency:** As the boards are taken from TEO, they are not fully suitable for this application and therefore consist of several superfluous elements. The new cabling will be specific to the neck, which will allow it to be smaller and meet the compactability condition.

Based on these considerations, it was decided to design a different type of connection board, obtaining in the process a board that will be manufactured and used in the integration.

3.5.2.2 Single board approach and final design

Finally, it was decided to look at the design of the connection boards from a different perspective. The idea was to maximise the space available for connections and component

placement. A connection plate is therefore proposed whose geometry coincides with that of the base it supports, resulting in a circular external perimeter.

It was decided to design a triple symmetry with a 120º angle for the drive and motor connections. On the other hand, unlike the previous version, there is only one connector for the power supply to the drive and the CANbus cable.

In this case, the CAN network is configured on the board itself, connecting all the CAN high pins and all the CAN low pins. On the other hand, in the redesign, it is proposed to have two pins for the introduction of a CAN termination resistor between the CAN High and CAN Low lines, in order to close the CAN network in experiments external to the humanoid.

Another new feature compared to the previous version is the presence of two additional copper planes, one for power and one for ground. This means that all pins that need to be connected to power or ground are routed directly to these planes without the need for additional tracks. A second advantage of having these planes is that it significantly reduces possible interference or noise in the signals on the board. In addition, a Molex power connector is added to ensure that higher voltages are passed to the board.

As the new connection boards had to be considerably smaller than the original ones, this idea was taken into account in the choice of drive model by selecting the smallest iPOS model with CANopen communication from the Technosoft product range. This is the iPOS 2401 MX-CAN, which is highly compatible with the system of the platform as it is from the same manufacturer as the iPOS 4808 MX [74], which also enables communication with the CANopen protocol.

The features of this new iPOS model are shown in Table 3.7.

Table 3.7: Electrical and communication properties of the iPOS 2401 MX-CAN driver from Technosoft.

Power	Motor and logic power source	Output current	Communication Protocols
24 W	12-24 V	0.9 A	TMLCAN/CANopen

Taking into account all these considerations, the resulting connection board is shown in Figure 3.27.

Figure 3.28 shows the final design of the soft neck ready for its integration in the humanoid robot TEO. As the base of the neck has three holes for the passage of three screws, these holes should also appear on the connector board, and will also be used for the attachment of the connector board to the base. It will also be necessary to include the central hole to allow the

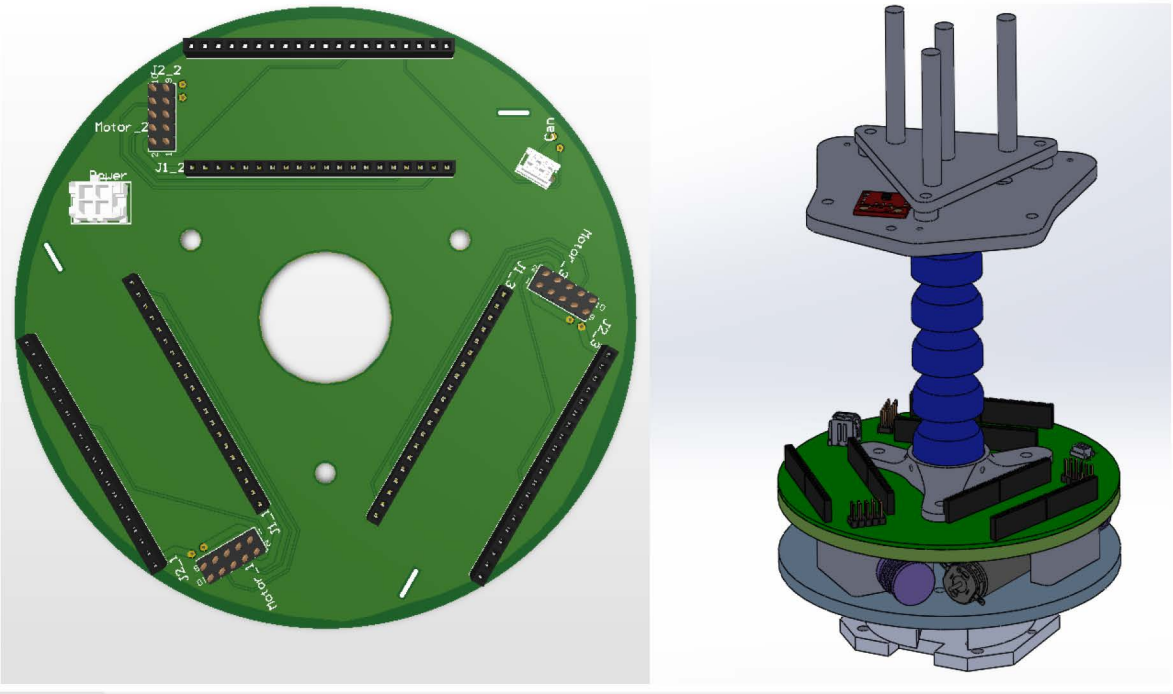


Figure 3.27: Final design of the connection board and CAD simulation of its assembly on the soft neck platform.

passage of cables between the bottom and top plates of the platform, mainly for the IMU sensor, which is located in the top base. Finally, orifices have been made for the passage of the wires that pull the top base.

3.6 Soft link-based arm

The main objective of the HumaSoft project is to design a versatile soft link to replace different functional rigid parts of the humanoid robot TEO. Once the soft neck design and development has been addressed, the focus is put now on the use of all this research for the design of a soft link able to act as a manipulator or arm for TEO (or any other mobile or static platform). The conceptual design already proposed will be reused and adapted for the correct performance of the manipulator.

The goal of this section is to design and prototype a tendon-driven soft manipulator with two DOF: inclination and orientation. A structure will be created that allows the arm to bend in different directions from its vertical axis. The soft arm will be actuated by motors that shorten or lengthen the internal tendons. This will result in a combination of actions and distances that will cause the arm to achieve different angles of inclination and orientation. A test bench for the manipulator will also be designed and manufactured.

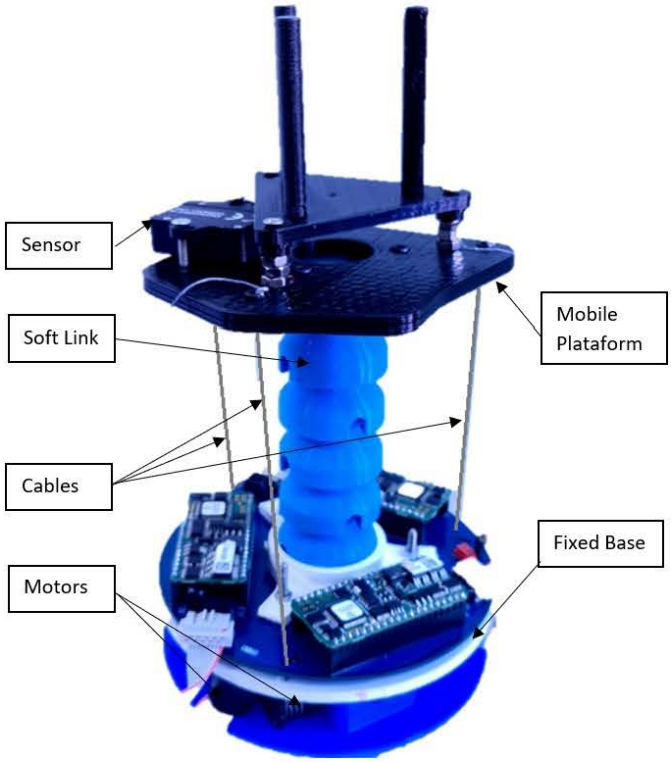


Figure 3.28: Overview of the redesigned cable-driven soft robotic neck.

It is desired that the soft manipulator rests horizontally, perpendicular to the gravitational force, unlike the previous model of the neck (Section 3.4), which rests in a vertical position, in the direction of the gravity force. This new arrangement means that the arm will need to be flexible but, at the same time, more robust than the neck version.

3.6.1 Mechanical system

The approach presented in this section is inspired by the soft robotic neck previously designed (Section 3.4). Differently from that design, this new solution consists of a novel three-dimensional soft joint morphology based on asymmetric links. The joint is composed of a soft material that is flexible but robust. This material allows deformation to achieve bending movements, but prevents too complex deformations and undesired gravity effects. When the designed soft joint is bending, from a certain inclination angle and at certain orientation angles given by model measurements, a natural protection is provided by its own morphology, which limits maximum bending. Therefore, sufficient robustness is achieved to support different loads throughout its positioning range in 3D space, while maintaining the advantages of its soft nature.

Furthermore, the proposed joint is scalable and adaptable to operational requirements in a modular and simple way. Therefore, joint properties, such as maximum bending angle or

blocking bending, can be configured by modifying the morphological design and number of the links in the joint, or the distance between them, as well as increasing the number of DOF by concatenating joints.

Finally, this proposal is a low-cost construction, primarily designed by 3D printing and actuated by three motors that vary the length of tendons. Tendons are integrated within the morphology itself, which favors constant curvature and simplification of the model. Electromechanical action is proposed for the articulation, as opposed to other energy sources such as pneumatics or hydraulics. This feature allows the portability of the prototype and a greater integrability in any system (a robot, a humanoid, etc.), as well as more precise control and easier maintenance.

3.6.1.1 Design and prototyping of the soft arm

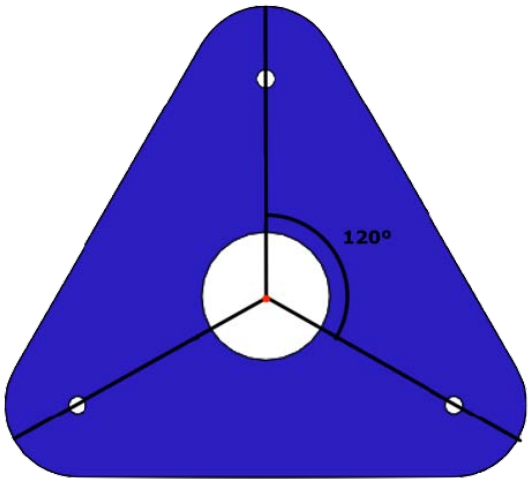
The soft joint has an asymmetrical morphology that allows its end tip to be positioned in the three-dimensional environment, robustly supporting high loads during its performance. Its design provides greater flexibility and a wider range of movement than a rigid joint. It consists of a series of links with asymmetrical prism morphology and circular section pitch. A triangular morphology is represented in Figure 3.29.

The small section and soft nature of the central axis of action allow a greater bending capacity in all directions. The asymmetrical prismatic section provides the property of blocking and a natural protection, as well as the routing of the tendons for their action.

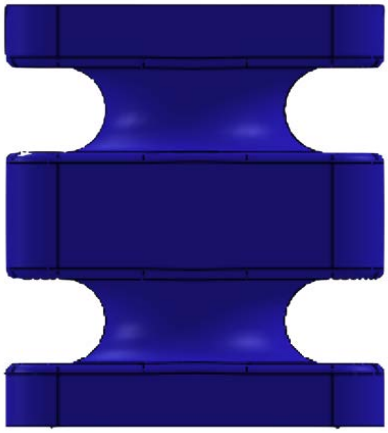
The design performance is achieved by tendons that are routed through the asymmetric prismatic sections, as shown in Figure 3.30. It is possible to change the morphology of the prism and route the tendons through different points of these sections. This change would cause the variation of the forces and moments the joint is subjected to, therefore obtaining different kinematics and dynamics. By acting on the tendons, the joint can bend and orientate with two DOF.

One of the novel characteristics of this design is the natural morphological protection of the joint against large loads provided by the proposed asymmetrical morphology. An example of the triangular morphology are the two different configurations of extreme load:

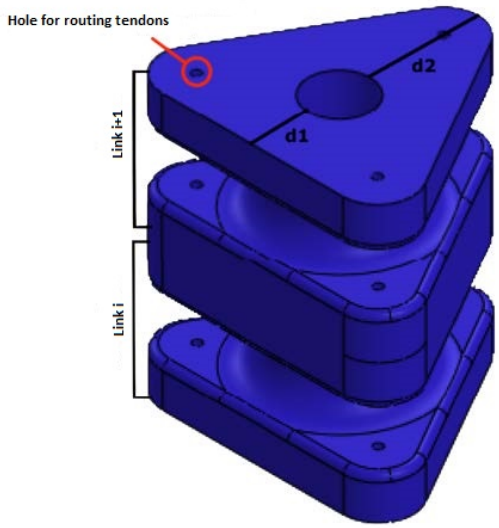
- Configuration 1: Flexion towards one of the vertices of the triangle.
- Configuration 2: Flexion towards one of the edges of the triangle.



(a) Top view showing the 120º angle relationship between the different tendon routing points.



(b) Front view.



(c) Perspective view, showing d1 and d2 distances defining asymmetry and holes for routing tendons.

Figure 3.29: Triangular asymmetric geometry of the soft joint with two links in different views.

In Configuration 1, protection when turning in the direction of one of the vertices is the most restrictive, as shown in Figure 3.31a. In the case of excessive bending, caused by high loads placed at the tip of the joint or by control failures, the vertices contact each other. This produces a blocking curve of the structure that protects the joint from possible breakage due to wear or due to exceeding its elastic limit. This protection allows the joint to act with robustness and safety, especially in the regions of maximum flexion. In this configuration, the action is achieved by a single tendon, which is routed through the vertices that form the bending curve.

Configuration 2 allows a larger flexion of the joint, compared to Configuration 1, while

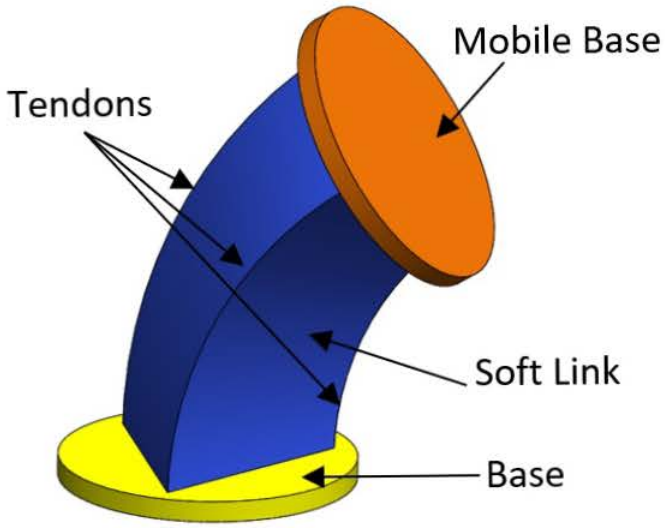


Figure 3.30: Conceptual design of the joint with its components: base, continuous soft axis, tendons for performance and tip (mobile base) of the soft joint.

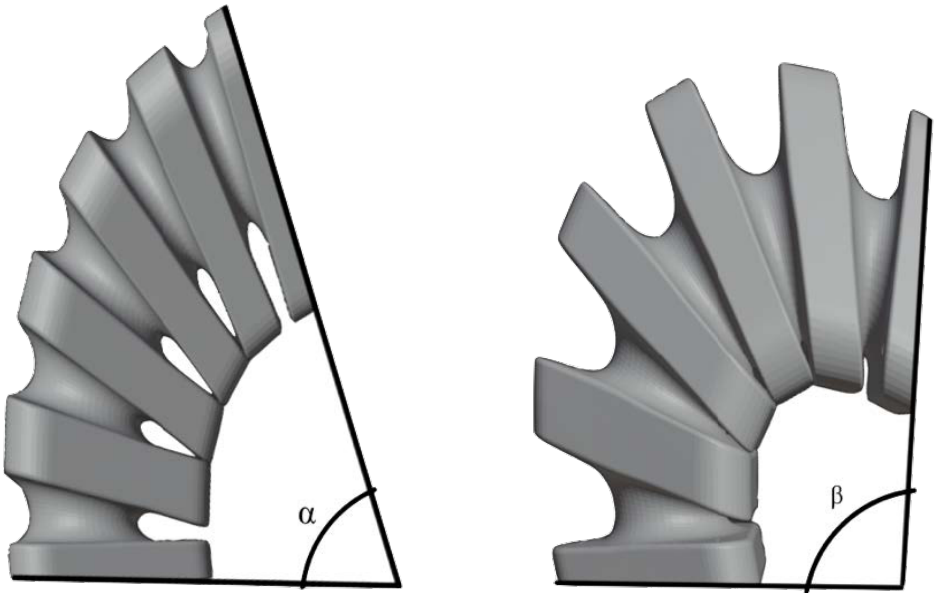
also maintaining the natural protection of the joint. When the flexion is towards one of the edges of the triangle, the blocking curve has a smaller radius, as shown in Figure 3.31b. This is because the edges are closer to the central axis of rotation, as can be seen from the distance ratio $d1 < d2$ in Figure 3.29c. A larger bending occurs due to the fact that a larger bending angle is necessary before these edges contact each other and lock the joint structure. In this configuration, the performance is achieved by the action of the two tendons that form the edge of the triangle where bending occurs.

3.6.1.2 Material properties and tests

One of the most important issues when prototyping a soft robotic joint is the choice of its material. This design uses again NinjaFlex® 3D Printing Filament, which, as described previously, is a flexible polyurethane material for Fused Deposition Modeling (FDM) printers. This 3D printing manufacturing method and this material are chosen for their ease to use and for allowing variations in percentage or filling patterns of the soft joint body.

The mechanical properties of this material make it a good choice for the purpose of the arm design (Table 3.6). Its flexibility allows the joint to bend but, at the same time, it is rigid enough to prevent big deformations and resist loads.

The soft joint design is analyzed in ©SolidWorks software, which applies a nonlinear finite element study on the material. The prototype is modeled as a simple cantilever beam (one of its ends is fixed and a force is applied to its free end). This allows an efficient testing of the design



(a) Flexion in configuration 1 has the lowest maximum bending angle.

(b) Flexion in configuration 2 has the highest maximum bending angle.

Figure 3.31: Different bending configurations. Relationship between bending angles: $\alpha < \beta$.

under stresses and strains.

To simplify the simulation, the joint is assumed to be a completely filled solid except for the inner channel, and to simulate the assembly of the real prototype, the soft joint model is assembled including its two support pieces, one at each end.

After the design phase, the prototype is 3D printed using NinjaFlex material with 30% infill. The experiments are performed with this specific prototype.

The model in ©SolidWorks is tested under different conditions. First, a no-load test is performed on the soft joint, by only simulating gravity and fixing one of the ends, as shown in Figure 3.32, with the red arrow representing the orientation of the gravity action in the simulation.

One intended use of this soft joint is as a manipulator able to support different loads. Therefore, a second simulation is carried out with a rectangular prism with a fixed mass of 500gr, homogeneously distributed. This prism represents the weight of the robot gripper in the simulation, Figure 3.33. In addition, a 10N downward force is applied to the end effector, simulating an external weight of 1kg and causing a higher end torque. The simulation shows a deflection of 7.38° and a maximum deformation of 0.75MPa.

Additionally, another stress study is carried out to check if the yield strength of Ninjaflex is not exceeded. It is noted that, when applying a 60N force at the end of the soft joint (Figure

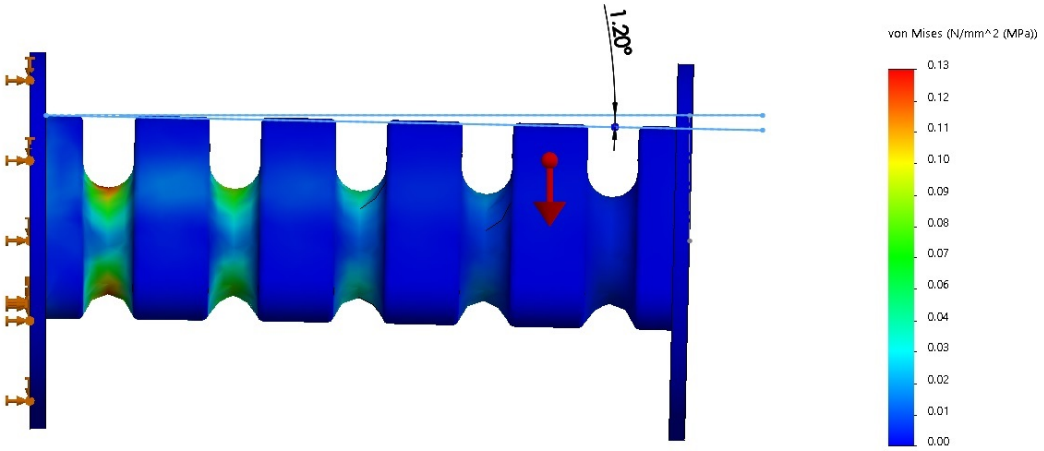


Figure 3.32: No-load simulation of the soft joint.

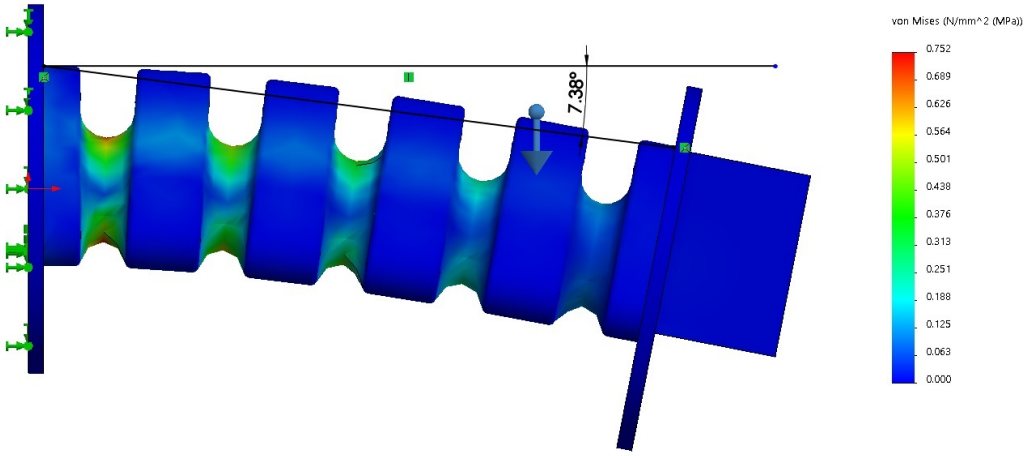


Figure 3.33: Simulation of the soft joint with a 500gr prism and a 10N downward force at the free end.

3.34), a bending angle of 60° is reached and the maximum deformation is of 2.9MPa. Therefore, a no permanent deformation is confirmed when the soft joint reaches an inclination angle of 60°.

3.6.1.3 Actuation

As mentioned above, there are several ways to operate soft robots. This desing focuses on operation by tendons of variable length using a winch coupled to a motor shaft. Tendon lengths must be translated into motor angular positions. L_o = 0.2m is the length of the tendons when the joint is at rest position, and L_i is the target tendon length. The linear displacement is transformed into an angular displacement by the length of the arc formed by the circumference of the winch for a certain angle (Figure 3.35), following the equation below:

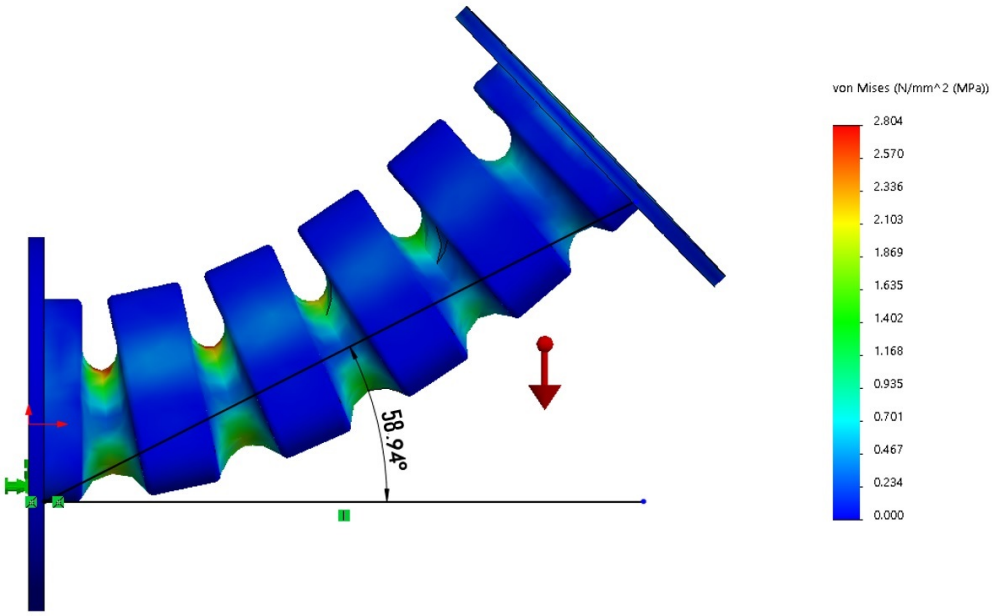


Figure 3.34: Simulation of soft joint with a 60N upward force applied on the free end.

$$\Omega = \frac{(L_o - L_i)}{R} \tag{3.25}$$

R is the radius of the winch where the tendon is wound or unwound, in this case 9.3mm, and Ω is the angle that provides that displacement.

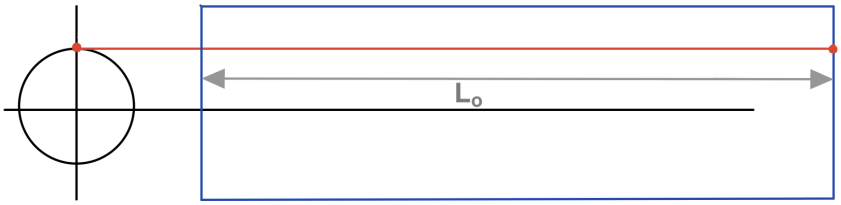
3.6.2 Electronic system

In this section, the motors and electronic components for the cable-actuated manipulator, which will act as a soft robotic arm, will be selected. The different components will be chosen based on the desired performance of the robotic arm.

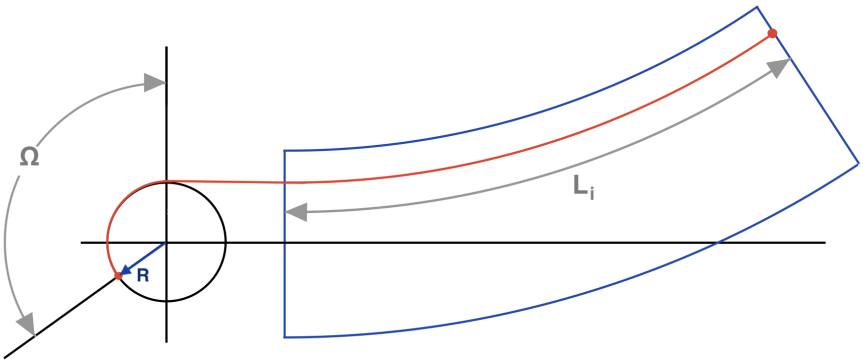
3.6.2.1 Motor selection

The selection of motors is achieved through an experimental approach. Using a weight meter, it is determined that approximately 60N of linear force is required to bend the arm 90º, as shown in Figure 3.36. This calculation is derived from equation $F = m \cdot g$, where F is the force, m is the mass and g is the gravity acceleration (rounded to 10). The balance indicates a mass of 6kg, which is used to determine the force required.

It is estimated that motors with a maximum torque of 0.6Nm could be used for this system. This is calculated using equation $T = d \cdot F$, where T is the motor torque, F is the force and d is



(a) Tendon and soft joint prior winch actuation.



(b) Tendon and soft joint when the winch is operating, with radius R and angle Ω.

Figure 3.35: Diagram depicting winch winding based on radius and angle. $L_o - L_i$ is the distance for tendon winding, and Ω and R are the angle and the radius, respectively.

the distance to the geometric axis of the motor. The force for this calculation is assumed to be 60N, as determined from the force analysis. In addition, a winch with a diameter of 2cm (d = 0.01m) is used to operate the motor and wind and unwind the tendon.

Using Maxon's selection programme [72], the desired motor data are introduced, with a supply voltage of 36V, a maximum load speed of 30rpm, a load torque of 0.6Nm, a diameter of less than 35mm and an encoder to control the position. The two proposed motor models are: Motor A-max 22 GB and Motor EC-max 22.

Motor A-max 22 GB: It is a brushed motor with a 198:1 planetary gearhead, capable of producing a load torque of 600mNm. Figure 3.37 shows that the motor is close to the performance limit required for the soft arm.

Motor EC-max 22: This motor has a maximum load torque of 800mNm, a brushless motor and Hall sensors, 157:1 planetary gear. As shown in Figure 3.38, the adjacent operating range is greater and will allow a better performance also when external loads are used on the soft arm.

The EC-max 22 has a brushless configuration and Hall sensors. Comparing their specifications, the EC-max 22 motor has a better behaviour in the working range we are looking for, it

Figure 3.36: Experimental measurement to obtain a hypothetical value of the necessary linear forces.

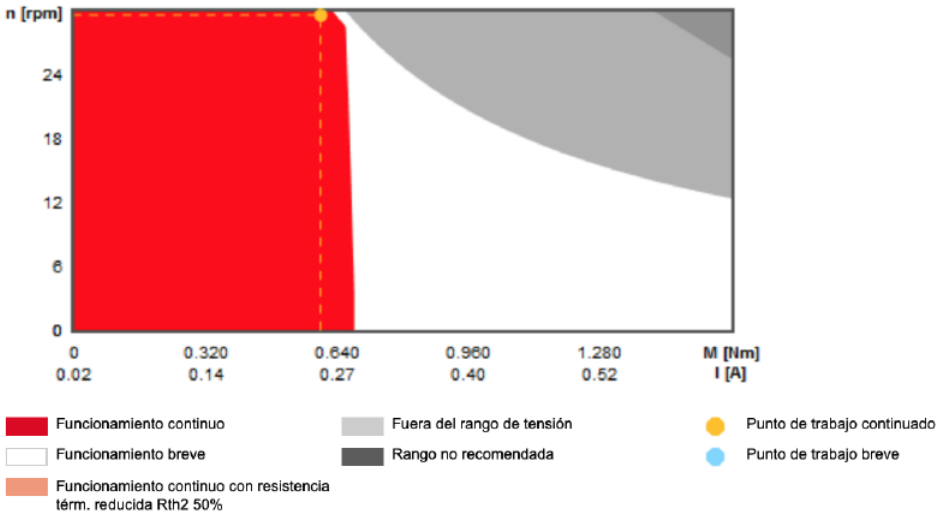


Figure 3.37: Motor A-max 22 GB performance graph comparing the revolutions per minute versus the moment and the current.

has a smaller gearbox and it is smaller in length. These are the reasons why this motor is finally selected for the soft arm prototype.

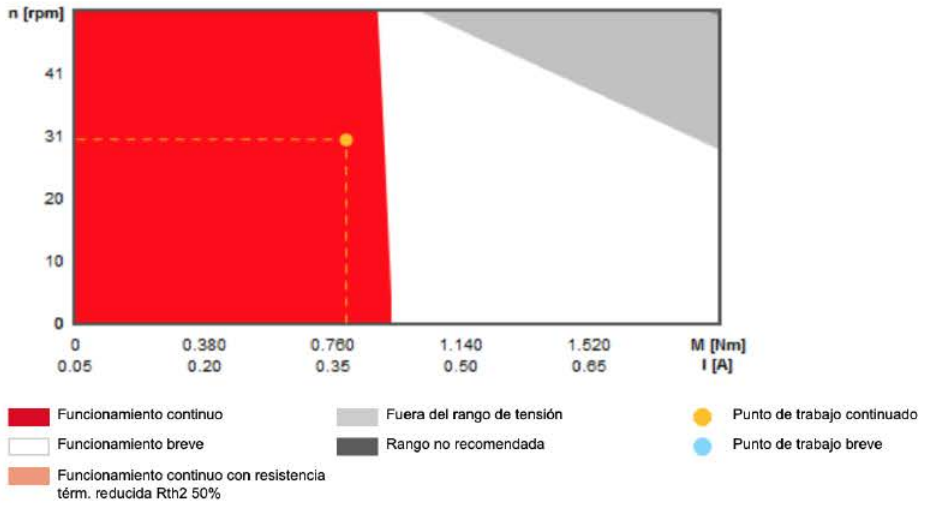


Figure 3.38: Motor EC-max 22 performance graph comparing the revolutions per minute versus the moment and the current.

3.6.2.2 Configuration of the electronic system

The IPOS intelligent driver allows the motors to interact with a computer to control the movements of the soft arm. This simplifies the design of the mechanism, as the previously selected and sized motors can be programmed to control the desired movements. This is achieved by specific commands designed and programmed into the IPOS driver. This allows the user to control the soft arm with greater precision and ease.

The electronic components used for the design of the platform are:

- **PC and PeakCAN PCAN-minipci card.** The PC acts as the control unit in CANopen communication, sending and receiving messages from drives using programmed code. To enable PC node communication with drives via a CAN network, the PC must be equipped with a card that adapts ports to the network. In the HumaSoft laboratory, the PeakCAN PCAN-minipci card is used for this purpose.

- **Intelligent Drives iPOS 4808 MX from Technosoft.** The Technosoft iPOS family of intelligent drives is used to perform all basic motor control functions and other motion control tasks. The iPOS4808 MX-CAN drive, shown in Figure 3.39, is used to control the motors and is suitable for controlling brushless DC, brushless AC (vector control), DC brushed motors and step motors; it is also compatible with the CANopen protocol [33]. Two base plates, each designed to hold up to two iPOS4808 MX-CAN drives, are used to integrate the three iPOS drives that control the motors. These components are chosen for their availability and ability to use CANopen communication.

- **Power source unit.** The motors used in the prototype are powered at 36V, while the drives operate at 48V. The power source used is the PROMAX FAC-662B.

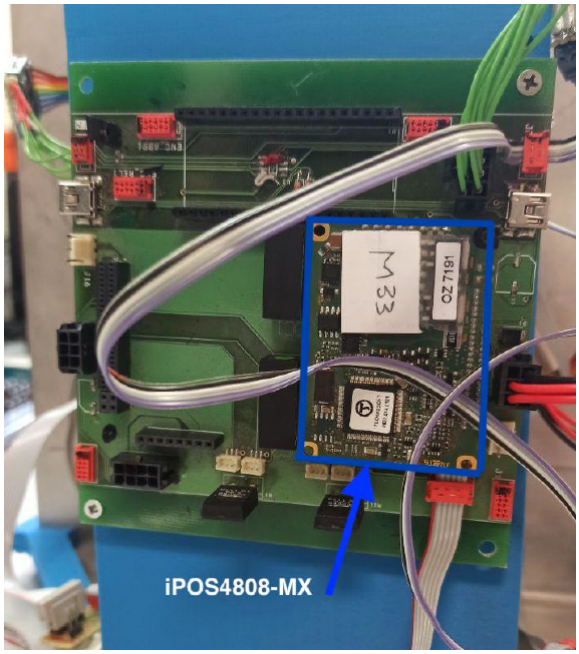


Figure 3.39: An iPOS4808 MX-CAN connected to the motherboard.

3.6.3 Prototype of the soft arm

The final soft arm prototype is shown in Figure 3.40, where the three motors connected to their tendons allow the actuation of the joint.

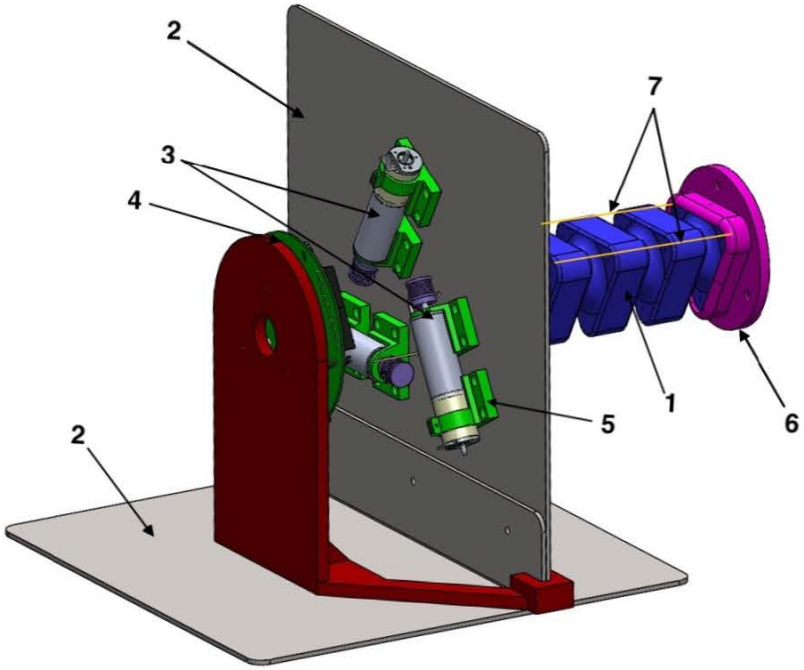
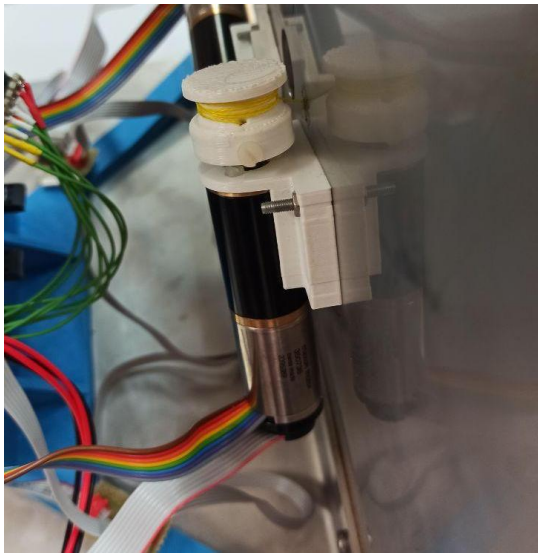


Figure 3.40: The elements of the platform are the soft joint (**1**), a metal base (**2**), motors (**3**), electronic elements to feed and control the motors (**4**) and other connective elements such as motor supports (**5**), joint bases (**6**) and tendons (**7**).

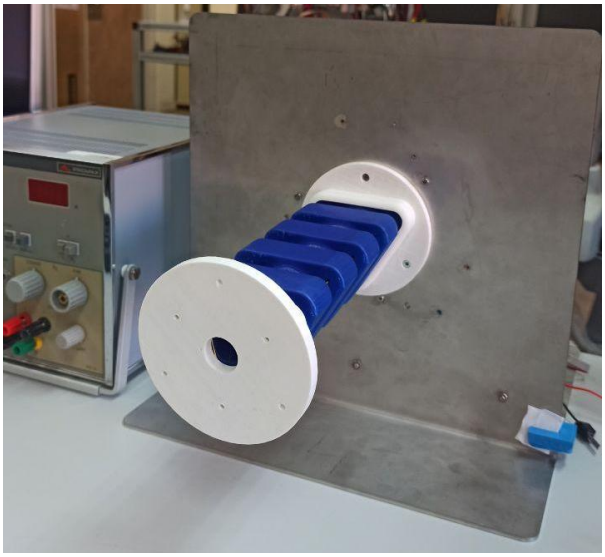
The fixing base is made up of two 3mm thick metal plates, to be strong enough to support the test loads. Connecting elements have been printed from PLA material using Creatbot600 pro and Zmorph 3D printers. These elements are two bases for attaching the soft joint to the metal base, a platform to attach the electronic elements, three connectors to attach the motors to the metal platform and three winches that are attached to the motor shafts and the tendons, made of polyester thread, for the activation of the joint (Figure 3.41).

3.6.4 Patent

During the development of this desing, a patent has been filed with the Oficina Española de Patentes y Marcas (OEPM) entitled "Eslabón para articulación blanda y articulación blanda que comprende dicho eslabón" (Link for soft joint and soft joint comprising such link), the soft arm design being one of the preferred embodiments of the invention. All the patent details are presented in Table 3.8. The licensing process is still running.



(a)



(b)

Figure 3.41: (**a**) Motor connected to the winch to wind the tendons for joint actuation. (**b**) Soft joint on the test platform.

Table 3.8: Patent information.

Reference	5349ES
Application	P2020030726
Priority Countries	Spain
Date	14/07/2020
Holder	UC3M

3.7 Communication interface

This section describes the characteristics of the communication protocols used in the libraries developed for the HumaSoft project. These protocols have been designed to be used in different driver models that allow communication between devices as well as between the computer and the devices. In addition, the general organisation of the communication libraries is described using class diagrams, which facilitates the creation of different applications for the user to operate with the joints. This communication network allows the user to interact with the devices and to receive, send and process information easily and quickly.

3.7.1 CanOpen communication

CANopen is a high-level communication system based on the CAN standard [75] and, by extension, the OSI reference model. CANopen was originally designed for applications within the automotive industry, particularly for its control systems. However, it is now used in

a wide range of applications, including robotics. Its success in robotics is due to its ability to standardise embedded networks and its high degree of flexibility.

The communication model follows master/slave, client/server and producer/consumer procedures between different nodes, and therefore the identification and profiling of network devices is very important. This is done using the COB-ID, a 29-bit system that allows up to 127 devices to be distinguished and establishes the priority rules between nodes [76].

For the development of the software platforms, HumaSoft decided to implement a CANopen network, due to the great advantages it offers in robotics and automation applications, and to maximise compatibility with TEO's system, which also follows the CAN and CANopen standards.

The network connected by CAN buses follows the scheme shown in Figure 3.42, where each device is assigned a node, these being the three drives and the computer. The network consists of two closed bus lines, called CAN low and CAN high, separated by two 120 ohm resistors connecting each node to these two lines.

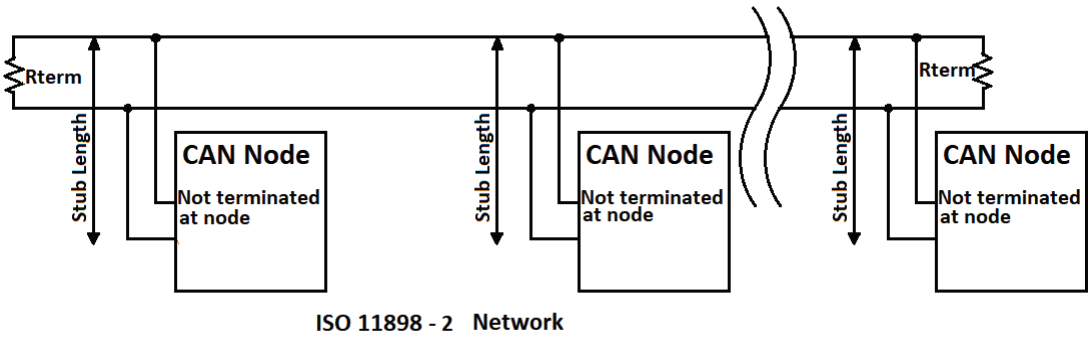


Figure 3.42: Schematic of a High Speed ISO 11898-2 network with limiting resistors between the CAN low and CAN High line [77].

The choice of different drives allows CANopen communication, so that their management can be carried out using this type of communication. In particular, these devices have a state machine that makes it possible to define the state of the drive and therefore the functionalities that the device can have. This state machine, shown in Figure 3.43, together with its control and status messages, is defined in the CiA402 CANopen Device Prole Drivers and Motion Control standard [76] and in the CiA402 Draft Standard Proposal [78].

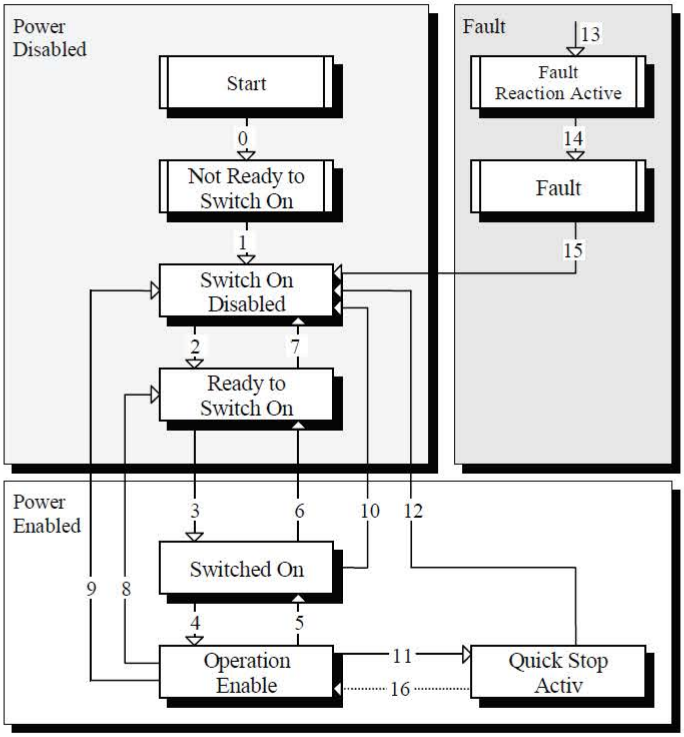


Figure 3.43: State machine of a device following the CANopen CiA402 standard [78].

3.7.2 Communication library

The device is controlled through the use of CANopen communication, for which communication libraries have been developed to allow messages to be sent between the drives and the control computer.

The organisation of the libraries follows a parallel structure to the CANopen standard, namely in the CiA402 [76] and CiA301 [79] standards. As explained in the previous section, the use of these protocols allows a simpler implementation of CANopen devices, as well as the possibility to replace them if another CANopen device is to be used. This choice has been fundamental for the software integration of the device, as it allows the use of any CANopen communication drive, as is the case in the previously designed prototypes, such as the iPOS 4808 MX drives for the spring-based neck and soft arm, and the iPOS 3604 MX used to integrate the soft neck in the humanoid robot TEO.

Therefore, the organisation of the libraries follows a hierarchical structure based on object-oriented programming in the C++ language, structured in classes with inheritance and usage relationships.

The lower level classes deal with port and data bus operations, such as writing to a port or clearing messages. Several higher level classes from the libraries inherit from this class.

One of them is the class that follows the CiA301 standard. This class describes simple functions that consist of sending and receiving the different types of CANopen protocol messages on the ports. This class also defines the object dictionary, or the communication protocols.

From this class, a higher level class is inherited according to the CiA402 standard. This class has more complex functionalities that use the functions of the lower classes to command a position or speed on a motor, or to set the operating modes and configuration of the drive.

Figure 3.45 shows the class diagram of the communication libraries and their main methods, inheritance and usage relationships.

The development process of these libraries and a compilation of the methods implemented in each of these classes is described in [77]. The latest versions of the HumaSoft communication libraries and their documentation can be found in the HumaSoft GitHub repository [80].

3.7.3 Importing C++ libraries to Python

Two years ago, HumaSoft was transformed into the SOFIA project with the specific goal of applying machine learning to soft robot joints. For this, communication libraries written in C++ were translated into Python using the Simplified Wrapper and Interface Generator (SWIG). This was achieved by creating common shared libraries for Python and C++. In this way, SWIG acts as an interface compiler that connects programs written in C++ to Python, as shown in Figure 3.44. The CANopen communication library must now also be used in Python.

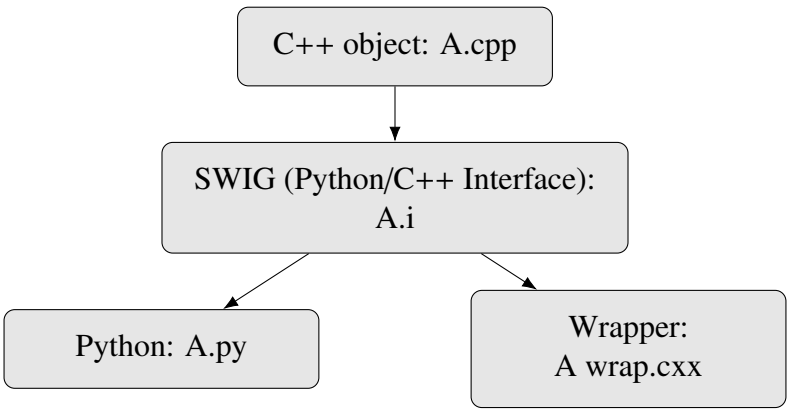


Figure 3.44: Interface through SWIG.

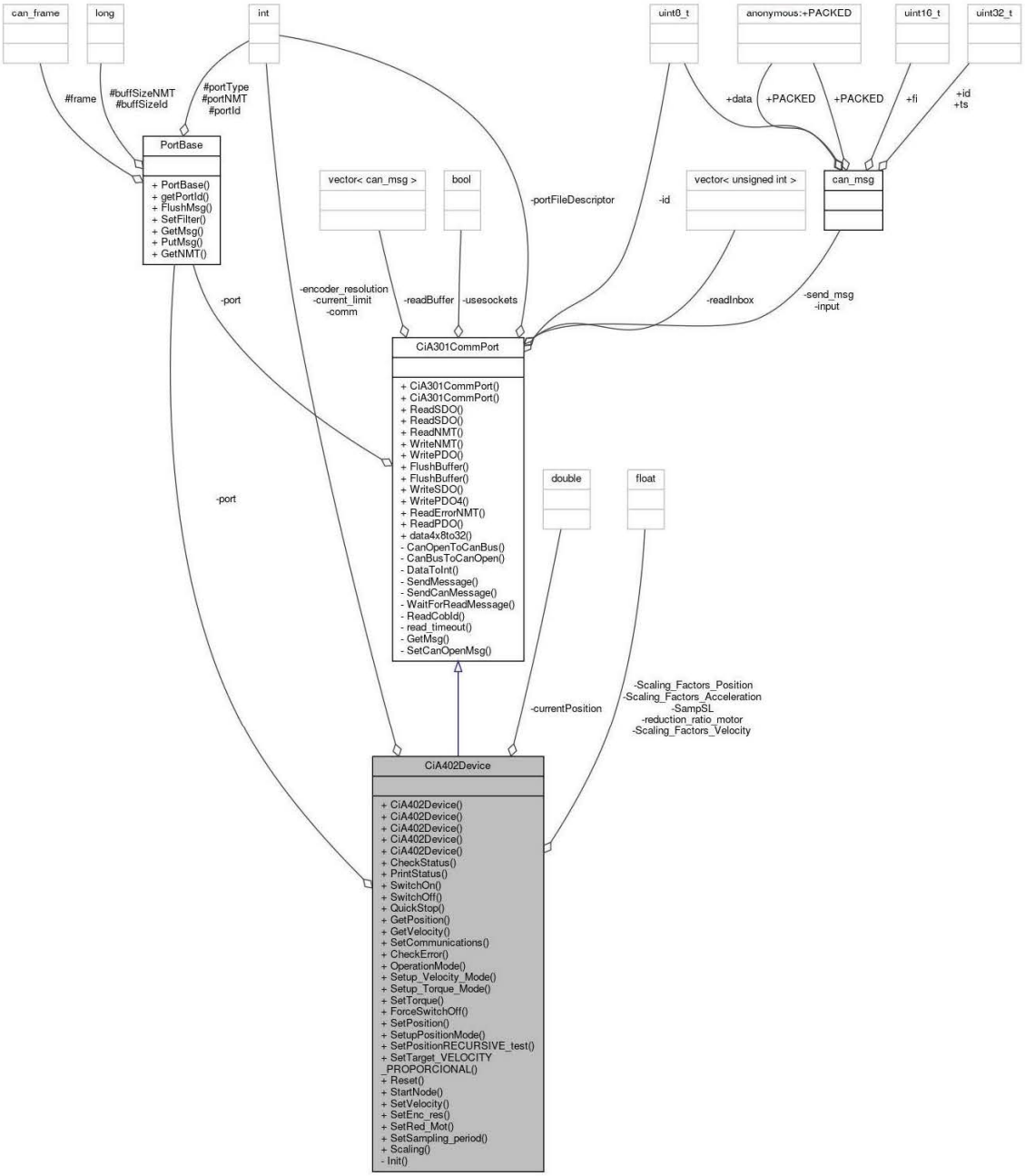


Figure 3.45: Class diagram of HumaSoft CANopen communication libraries [80].

3.8 Conclusions

In this chapter three different designs of soft joints have been proposed: a spring-based neck, a soft link-based neck and a soft arm. The three prototypes have been completely built after the definition of their different mechanical, electrical and communication systems. The modeling and testing of the soft arm are out of the scope of this work, since they are being addressed in another thesis. The modeling and testing of the neck prototypes will be addressed in the following chapters in order to validate their performance.

Chapter 4

ANALYTICAL MODELS OF THE SOFT JOINTS

This chapter describes in detail the mathematical modeling of the cable-driven parallel mechanisms described in the previous chapter, both for the spring-based neck and for the soft link-based neck. It addresses the analysis of the inverse kinematics and direct kinematics and the final simulation and validation of the models.

4.1 Analytical model of the spring-based neck

The analytical model of the spring-based neck is presented here, including the spring parameters and the inverse and direct kinematics. This model is developed with the goal of being used in control systems. The model will be first validated in simulation for verification. The results obtained will determine the potential of the model to be used in neck motion control applications.

4.1.1 Configuration of the mechanism

The structure of the cable-driven parallel robot, as shown in Figure 4.1, consists of four main components:

- **Cables.** Flexible cables of negligible mass and diameter are connected from the moving platform at points B_i ($i = 1, 2, 3$) to the fixed base at points A_i. The points A_i and B_i are equidistant from each other on a circle with radii $|OA_i| = a$ and $|OB_i| = b$, respectively,

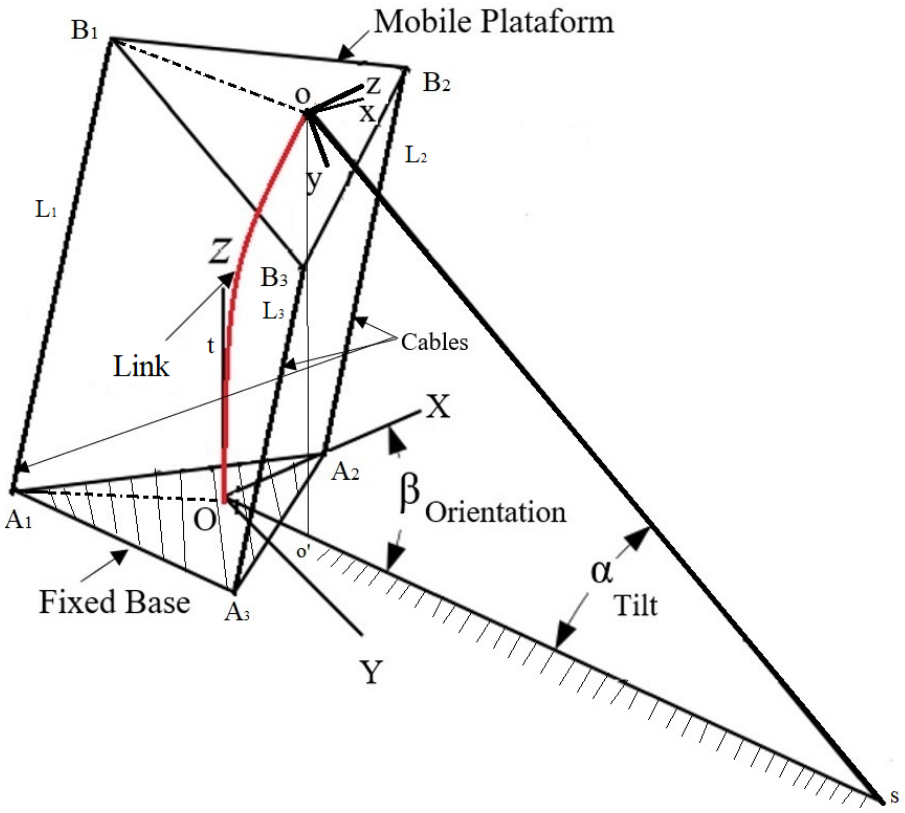


Figure 4.1: Schematic diagram of the CDPM.

with respect to the centre O and o. The vectors OA_i and OB_i are in the same direction in the initial configuration. The length of the wire between A_i and B_i is denoted as L_i.

- **Fixed base.** The origin of the fixed $OXYZ$ coordinate system is at the bottom centre of the spring. The Y axis extends along OA_1 while the Z axis is determined by the right hand rule. The homogeneous coordinates of the fixed base points A_i, as shown in Figure 4.2, are described as follows:

$$\overrightarrow{OA_i} = \begin{bmatrix} 0 & -\frac{\sqrt{3}}{2}a & \frac{\sqrt{3}}{2}a \\ a & -\frac{1}{2}a & -\frac{1}{2}a \\ 0 & 0 & 0 \end{bmatrix} \tag{4.1}$$

- **Mobile platform.** The moving part of the mechanism to which the load will be attached is fixed to the $oxyz$ coordinate system. The origin o is at the top centre of the spring, the y axis is along oB_1 and the z axis is determined using the right hand rule. The homogeneous coordinates of B_i (Figure 4.3) are inside the moving platform with respect to the $oxyz$ coordinate system.

$$\overrightarrow{oB_i} = \begin{bmatrix} 0 & -\frac{\sqrt{3}}{2}b & \frac{\sqrt{3}}{2}b \\ b & -\frac{1}{2}b & -\frac{1}{2}b \\ 0 & 0 & 0 \end{bmatrix} \tag{4.2}$$

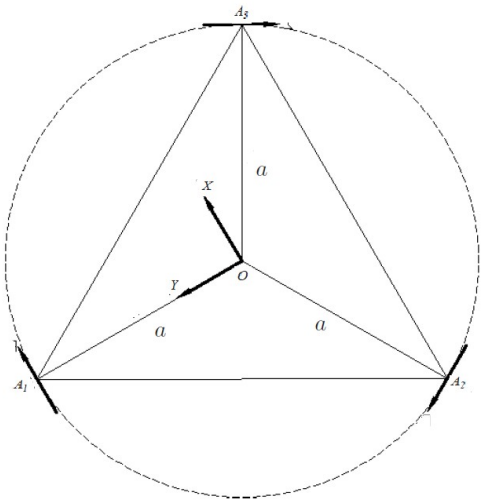


Figure 4.2: Top view of the fixed base.

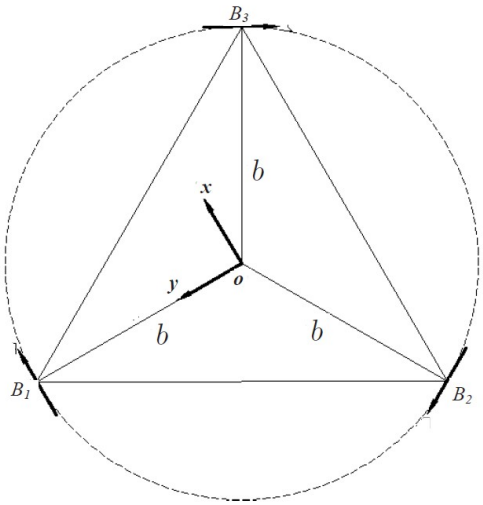


Figure 4.3: Top view of the mobile platform.

- **Spring.** The compression spring creates a force between the fixed base and the moving platform to support the load and allow the robot neck to move. It is shown as a curve from point O to point o in Figure 4.1. Note that the tangent vector of the spring curve at O is perpendicular to the plane of the fixed base, since it is rigidly attached to the fixed base at point O. For point o and the plane of the moving platform, similar arguments can be applied.

Assuming the cables are symmetrical at 120°, the spring is assumed to bend in one plane. The torsional resistance of the spring compression is quite high, so it can be assumed that the platform will not move around the z axis of the $oxyz$ reference frame. This fact will be taken as an assumption from now on.

4.1.2 Condition of the mechanism

The plane for the spring bending is shown in Figure 4.1, which is formed by the points O and o', where o' is the vertical projection onto the fixed base. The flexed spring has been projected on the plane formed by the coordinates Ost. The origin of this plane coincides with the fixed $OXYZ$ coordinate system, the t axis is aligned with the Z axis of the $OXYZ$ coordinate system and the s axis extends from point O to point o'. Four parameters are required to define the configuration of the mobile platform under these conditions.

As shown in Figures 4.1 and 4.4, the configuration of the moving platform is determined by four parameters: β (orientation angle), the angle between the s-axis and the x-axis; α (inclination angle), the angle between the fixed base plane and the plane of the moving platform; t_0, the vertical length of the bending spring; and s_o, the lateral translation of the bending spring. There are three independent parameters, with s_o being considered as a dependent parameter. This means that once β, α and t_0 have been determined, s_o can be determined. In this case, s_o is referred to as the parasitic motion [81], which can be calculated on the base of the other three parameters.

Once both β and α are known, the rotation matrix of the coordinate system $oxyz$ around $OXYZ$ can be determined, where β represents the orientation and α the inclination of the moving platform.

4.1.3 The rotation matrix

One of the fundamental problems in finding the 3D orientation workspace is the choice of coordinates to describe the orientation of the mobile platform.

The Euler parameters [82], direction cosines and other redundant sets of orientation coordinates provide a global parameterisation of orientation, but require a representation in 4D space. To avoid this, three Euler angles can be used to represent the orientation of a moving platform. These angles refer to three or more successive rotations around the base and axes of

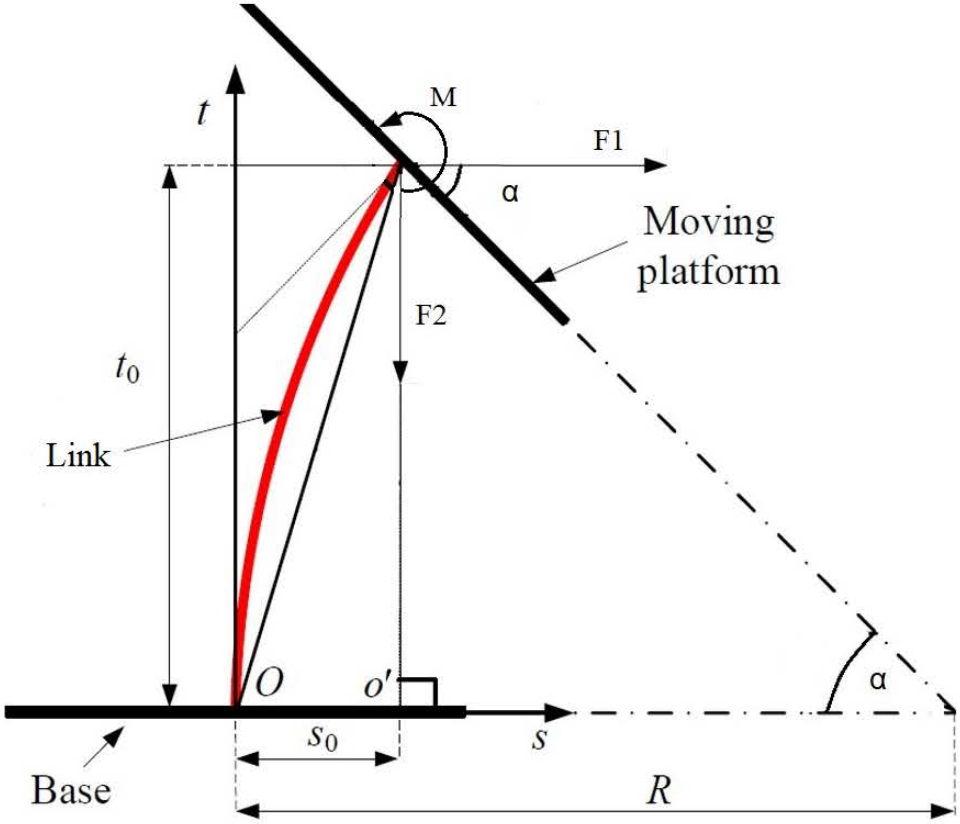


Figure 4.4: Lateral bending of the platform.

the platform. However, this method has the disadvantage that there are singularities where there is no direct correspondence between the actual orientation and the Euler angles.

The standard Euler angles meet the first requirements for Euler angles. This choice of Euler angles is defined by rotating the moving frame around the base z-axis an angle ϕ, then around the moving y'-axis an angle θ, and finally around the moving z'-axis an angle ψ. The singularity of this choice of angles occurs at $\theta = 0^{\circ}$, and the associated rotation matrix is:

$$R = R_z(\phi)R_{y'}(\theta)R_{z'}(\psi) = R_z(\psi)R_y(\theta)R_z(\phi) \tag{4.3}$$

In [83] a common modification of Euler angles was introduced to simplify orientation workspace plots for parallel manipulators. This new representation involves rotating the moving platform around the base z axis by an angle ϕ, then around the base y axis by an angle θ, followed by another rotation around the z axis by an angle ϕ, and finally around the moving z' axis by an angle ψ. The singularity is found at $\theta = 0^{\circ}$. As shown in Figure 4.5, ψ is the rotation angle, θ is the tilt angle, and ϕ is the angle between the base axis x and the projection of the approximation vector onto the base plane xy.

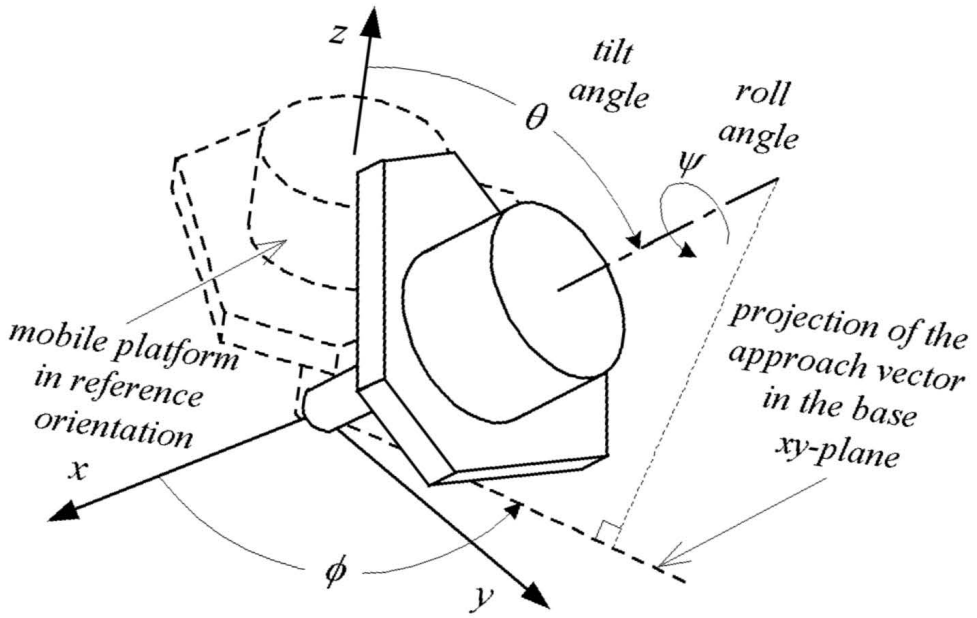


Figure 4.5: Modified Euler angles defining the platform orientation [83].

Assuming there is no rotational torque, the rotation angles ψ must be equal to zero. Therefore, the rotation matrix is defined as:

$$\begin{aligned} R =& R_z(\phi)R_y(\theta)R_z(-\phi)R_{z'}(\psi) \\ =& R_z(\phi)R_y(\theta)R_z(\psi - \phi) \\ =& R_z(\phi)R_y(\theta)R_z(-\phi) \end{aligned} \tag{4.4}$$

The rotational transformation is defined as:

$$\begin{aligned} R_x(\theta) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \quad R_y(\phi) = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix} \\ R_y(\gamma) &= \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{4.5}$$

Then, equating (4.5) to (4.4), the result is:

$$R(\phi,\theta) = R_z(\phi)R_y(\theta)R_z(-\phi)$$

$$= \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.6}$$

$$= \begin{bmatrix} \left(\cos^2\phi\cos\theta + \sin^2\theta\right) & (\cos\phi\cos\theta\sin\phi - \sin\phi\cos\phi) & (\cos\phi\sin\theta) \\ (\sin\phi\cos\theta\cos\phi - \cos\phi\sin\phi) & \left(\sin^2\phi\cos\theta + \cos^2\phi\right) & (\sin\phi\sin\theta) \\ (-\sin\theta\cos\phi) & (\sin\theta\sin\phi) & \cos\theta \end{bmatrix}$$

By substituting $\phi = \beta$ and $\theta = \alpha$ into Equation (4.6), the rotational transformation matrix of the robotic neck can be obtained:

$$^{O}R_{o'} = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix} \tag{4.7}$$

where:

$$t_{11} = \sin^2\beta + \cos\alpha\cos^2\beta$$

$$t_{12} = t_{21} = (\cos\beta - 1)\cos\beta\sin\beta$$

$$t_{13} = -t_{31} = \sin\alpha\cos\beta$$

$$t_{22} = \cos^2\beta + \cos\alpha\sin^2\beta$$

$$t_{23} = -t_{32} = \sin\alpha\sin\beta$$

$$t_{33} = \cos\alpha$$

As a result, the homogeneous transformation matrix between the moving base and the fixed base coordinate system can be expressed as:

$$^{O}T_{o'} = \begin{bmatrix} ^{O}R_{o'} & ^{O}P_{o} \\ 0 & 1 \end{bmatrix} \tag{4.8}$$

where $^{O}P_{o}$ is the position vector of point o with respect to the base coordinate system, given by:

$$P_o = \begin{bmatrix} s_o\cos\alpha & s_o\sin\alpha & t_0 \end{bmatrix}^T \tag{4.9}$$

4.1.4 Inverse kinematics

The analysis of the inverse kinematics of the robotic neck is represented by the following diagram:

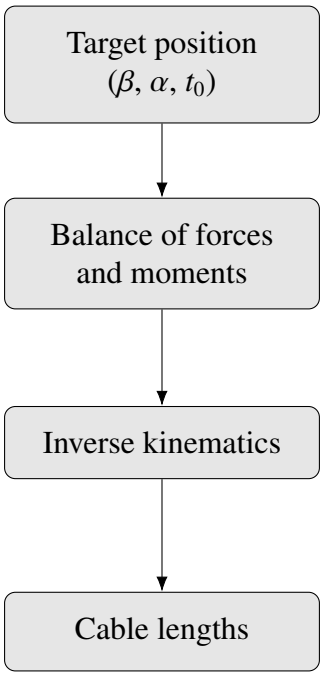


Figure 4.6: Solution of the inverse kinematic model of the robotic neck.

The inverse kinematics for the neck position problem can be described as follows: given the target position of the mobile platform, $x = [\beta, \alpha, t_0]^T \in \mathbb{R}^3$, the length of the cables, $L = [L_1, L_2, L_3]^T \in \mathbb{R}^3$, needs to be calculated, which can be described as:

$$L = f(x) \qquad f : \mathbb{R}^3 \longrightarrow \mathbb{R}^3 \tag{4.10}$$

The cable lengths are calculated by:

$$L_i = \left\| {}^{O}T_{o'} \overrightarrow{oB_i} - \overrightarrow{OA_i} \right\| \quad (i = 1, 2, 3) \tag{4.11}$$

It is not possible to arbitrarily assign s_0 to the solution of x due to the lateral bending of the spring caused by the forces acting on it. These forces are mainly related to the tensile forces of the three cables and the mass of the payload. Therefore, a combination of inverse kinematics and static analysis is required to obtain a solution of s_0.

4.1.4.1 Force-moment equilibrium equations

Section 4.1.2 describes how all forces in the cable can be transformed to the bending plane Ost. This is achieved by transforming all forces into two perpendicular forces F_1 and F_2 and a moment M perpendicular to the plane at the top centre of the spring (see Figure 4.4). Note that the motion of the platform is not subject to any external forces [84].

The equilibrium conditions for force and moment of the mobile platform are as follows:

$$\sum_{i=1}^{3} {}^{O}T_i + F = 0 \tag{4.12}$$

$$\sum_{i=1}^{3} {}^{O}r_i \times {}^{O}T_i + M = 0 \tag{4.13}$$

where:

$$\begin{aligned}
{}^{O}T_i &= T_i({}^{O}T_{o'}\overrightarrow{oB_i} - \overrightarrow{OA_i}) / \left\| {}^{O}T_{o'}\overrightarrow{oB_i} - \overrightarrow{OA_i} \right\| \\
{}^{O}r_i &= {}^{O}R_{o'} \cdot \overrightarrow{oB_i} \\
F &= \begin{bmatrix} -F_1\cos\alpha, & -F_1\sin\alpha, & F_2 - mg \end{bmatrix}^T \\
M &= \begin{bmatrix} -M\sin\alpha, & M\cos\alpha, & 0 \end{bmatrix}^T
\end{aligned}$$

Equations (4.12) and (4.13) presented in [84] can be decomposed into six equations containing a total of seven unknowns: T_1 to T_3, F_1, F_2, M and s_0. Eliminating T_1 to T_3 gives a single equation from which it is possible to obtain the values of F_1, F_2, M and s_0:

$$\begin{aligned}
&+2b\sin\alpha\sin\beta F_2' s_0^2 + 2b\left(t_0\sin\alpha\sin\beta F_1 + t_0\sin\alpha\cos\beta F_2'\right. \\
&\left. + \sin\alpha\sin\beta M + \frac{1}{2}a\sin\beta\cos 2\alpha F_2'\right) s_0 \\
&+b\left(2t_0^2\sin\alpha\cos\beta - ab\sin\alpha\sin^2\beta + at_0\sin\beta\cos 2\alpha\right) F_1 \\
&-ab\sin\alpha\sin\beta\left(a - b\cos\beta\right) F_2' - 2t_0\sin\alpha\left(a - b\cos\beta\right) M = 0
\end{aligned} \tag{4.14}$$

where:

$$F_2^{'} = F_2 - mg$$

4.1.4.2 Lateral bending equations for a spring

The first investigation of spring bending was made by Timoshenko, as mentioned in Section 3.3.1.1. It was suggested that the same methods as for elastic bars would work for the compression spring, although changes in length due to compression, which are not negligible, should be taken into account [68]. This proposal is shown in Figure 4.4. The spring is bent by the forces F_1 and F_2 plus a torque M and the deformation is described by the following equation:

$$\beta_s \frac{\frac{d^2 s}{dt^2}}{\left[1 + \left(\frac{ds}{dt}\right)^2\right]^{3/2}} = M + F_2(s_0 - s) + F_1(t_0 - t) \tag{4.15}$$

Equation (4.15) represents the total moment applied to a spring-cross section. It has no analytical solution, so it is necessary to calculate a numerical solution using elliptic integrals [85]. For small bending, the case can be simplified to a linear equation:

$$\beta_s \frac{d^2 s}{dt^2} = M + F_2(s_0 - s) + F_1(t_0 - t) \tag{4.16}$$

The end of the spring connected to the fixed base was defined as the initial condition, while the free end was connected to the moving platform:

$$s(0) = 0,\ s'(0) = 0,\ s(t_0) = s_0,\ s'(t_0) = \tan\alpha \tag{4.17}$$

where:

$$s' = ds/dt.$$

Based on Equations (4.16) and (4.17), two equations can be derived with respect to F_1 and M as a function of the parameter s_0:

$$F_1 = D_1 s_0 + E_1 \tag{4.18}$$

$$M = D_2 s_0 + E_2 \tag{4.19}$$

where:

$$D_1 = -\frac{a_2 c_1 - a_1 c_2}{a_2 b_1 - a_1 b_2} \qquad E_1 = \frac{a_2 d_1 - a_1 d_2}{a_2 b_1 - a_1 b_2}$$

$$D_2 = -\frac{b_2 c_1 - b_1 c_2}{b_2 a_1 - b_1 a_2} \qquad E_2 = \frac{b_2 d_1 - b_1 d_2}{b_2 a_1 - b_1 a_2}$$

$$a_1 = 1 - \cos\left(\sqrt{F_2/\beta_s}\, t_0\right)$$

$$b_1 = \sqrt{\beta_s/F_2}\, \sin\left(\sqrt{F_2/\beta_s}\, t_0\right) - t_0 \cos\left(\sqrt{F_2/\beta_s}\, t_0\right)$$

$$c_1 = -F_2 \cos\left(\sqrt{F_2/\beta_s}\, t_0\right)$$

$$d_1 = 0$$

$$a_2 = \sqrt{F_2/\beta_s}\, \sin\left(\sqrt{F_2/\beta_s}\, t_0\right)$$

$$b_2 = \cos\left(\sqrt{F_2/\beta_s}\, t_0\right) + t_0 \sqrt{\beta_s/F_2}\, \sin\left(\sqrt{F_2/\beta_s}\, t_0\right) - 1$$

$$c_2 = F_2 \sqrt{F_2/\beta_s}\, \sin\left(\sqrt{F_2/\beta_s}\, t_0\right)$$

$$d_2 = -F_2 \tan\alpha$$

Replacing Equations (4.18) and (4.19) in Equation (4.14), according to [84], yields the following equation:

$$A s_0^2 + B s_0 + C = 0 \tag{4.20}$$

where:

$$
\begin{aligned}
A =& 2b\sin\beta\sin\alpha\left(F_2^{'} + t_0 D_1 + D_2\right) \\
B =& \left(2bt_0^2\sin\beta\cos\alpha - ab^2\sin\beta\sin^2\alpha + abt_0\sin\alpha\cos 2\beta\right)D_1 \\
& - 2t_0\sin\beta\left(a - b\cos\alpha\right)D_2 + 2bt_0\sin\beta\left(F_2^{'}\cos\alpha + E_1\sin\alpha\right) \\
& + 2b\sin\alpha\left(E_2\sin\beta + \frac{1}{2}aF_2^{'}\cos 2\beta\right) \\
C =& \left(2bt_0^2\sin\beta\cos\alpha - ab^2\sin\beta\sin^2\alpha + abt_0\sin\alpha\cos 2\beta\right)E_1 \\
& - ab\sin\alpha\sin\beta\left(a - b\cos\alpha\right)F_2^{'} - 2t_0\sin\beta\left(a - b\cos\alpha\right)E_2
\end{aligned}
$$

Equation (4.20) is a quadratic equation and A, B, C are known if β and α are given. Then, s_0 can be obtained by solving Equation (4.20). Once s_0 is obtained, the inverse kinematics can be solved accordingly.

4.1.5 Simulation of the mathematical model

The inverse kinematics and static model is implemented in ©Matlab. The spring compression parameters for the prototype are shown in Table 3.1, and the bending stiffness constant is also given in Table 3.2.

Other parameters to consider are $a = b = 0.05m$ and $m = 1kg$. The implementation is done with a constant $t_0 = 0.1m$ because in real applications t_0 can only be used to adjust the force on the three cables. Varying α from 0° to 40° and β from 0° to 360° gives the results shown in Figure 4.7, where the cable lengths are in the z coordinate. This figure shows that, as long as α is large, the variation in length is also large. This is because the more inclined the moving platform is, the more force is required.

Figure 4.8 shows the system implemented in ©Matlab Simulink. In this model, all the blocks correspond to the subsystems described in Section 4.1.4. From left to right, we have: the target position block (where the necessary parameters are introduced to configure the mechanism), the static analysis block (state of equilibrium of the forces and moments of the mobile platform), the inverse kinematics block (calculation of the lengths of the cables for the parallel mechanism), the simulation graph block (graphical representation of the mechanism according

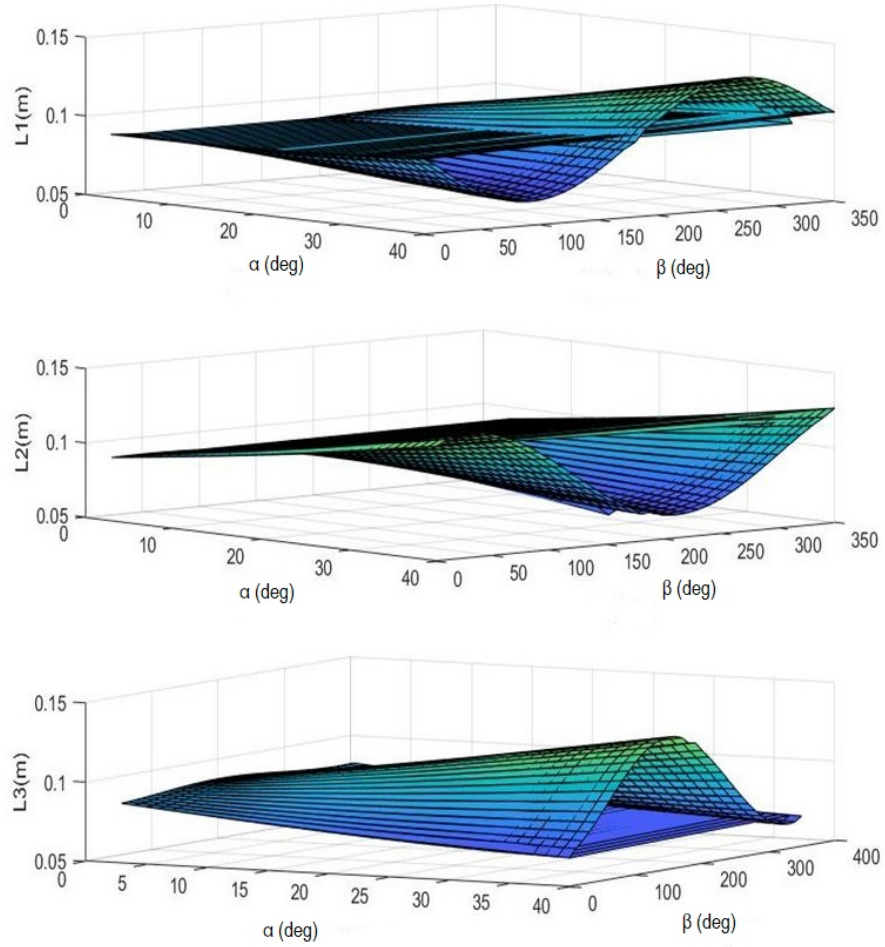


Figure 4.7: Inverse kinematics solution.

to the parameters introduced), DC motors blocks (each cable is connected to its motor and cable lengths are translated into motor angular positions).

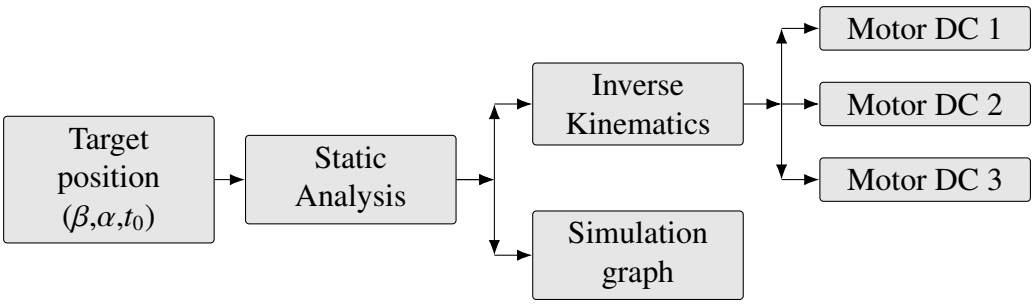


Figure 4.8: Block diagram for simulation of the robotic neck.

Figure 4.9 shows the ©Matlab Simulink implementation of the block diagram in Figure 4.8. The system has (from left to right) input signals (target positions), static analysis (where the output variables are F_1, F_2, M and s_0 described in Section 4.1.4), inverse kinematics model (giving as outputs the lengths of the tree cables), graph box (graphical simulation of the mech-

anism movement), a linear to angular displacement function for each motor (to convert the cable length to the motor reference position) and, finally, Motor 1,2,3 blocks (representing the dynamic model of the RE 273762 motors used for the real implementation of the neck).

Figure 4.10 shows the simulated three-dimensional CDPM in the rest position, and Figure 4.11 simulates its inclination and orientation movements.

It is also verified that the mechanism complies with the geometric parameters previously described in Section 3.1. The mobile base shall be tested to ensure that it is able to achieve a maximum inclination of 40° by operating each actuator individually, taking into account the length of each cable connected to its respective motor, as shown in Figure 4.12 (that maximum inclination is achieved with a negligible error).

4.2 Analytical model of the soft link-based neck

This section presents the analytical model of the soft link-based neck, where the spring is replaced by the soft continuum presented in the previous chapter. The inverse and direct kinematics of the robotic neck are addressed from the geometric distribution of the mechanism. Linear equations are also presented to allow an easy calculation of the lengths of the cables that determine the position of the neck in terms of inclination and orientation angles.

4.2.1 Inverse kinematics

The system of interest now in the one presented in Section 3.4, whose schematic diagram is the one shown in Figure 4.1. It has four main components: a fixed base with a coordinate system XYZ, a mobile platform with a coordinate system $oxyz$, three cables with negligible mass and diameter, and a soft link.

The mobile platform is driven by the cables connected to points B_i ($i = 1, 2, 3$). The opposite ends of the cables pass respectively through points A_i ($i = 1, 2, 3$) of the fixed base with coordinate frame $OXYZ$ and are connected to a DC motor. The X-axis and the Y-axis are along $\overrightarrow{OA_2}$ and $\overrightarrow{oB_2}$, respectively, and the z-Z axes are perpendicular to the fixed base and the mobile platform, respectively.

The soft neck has two independent pose parameters $x = [\beta, \alpha]^T \in \mathbb{R}^2$, the inclination angle α and the orientation angle β, as shown in Figure 4.1. To achieve the desired position, the cable lengths $L = [L_1, L_2, L_3]^T \in \mathbb{R}^3$ must be calculated as a function of the pose, as follows:

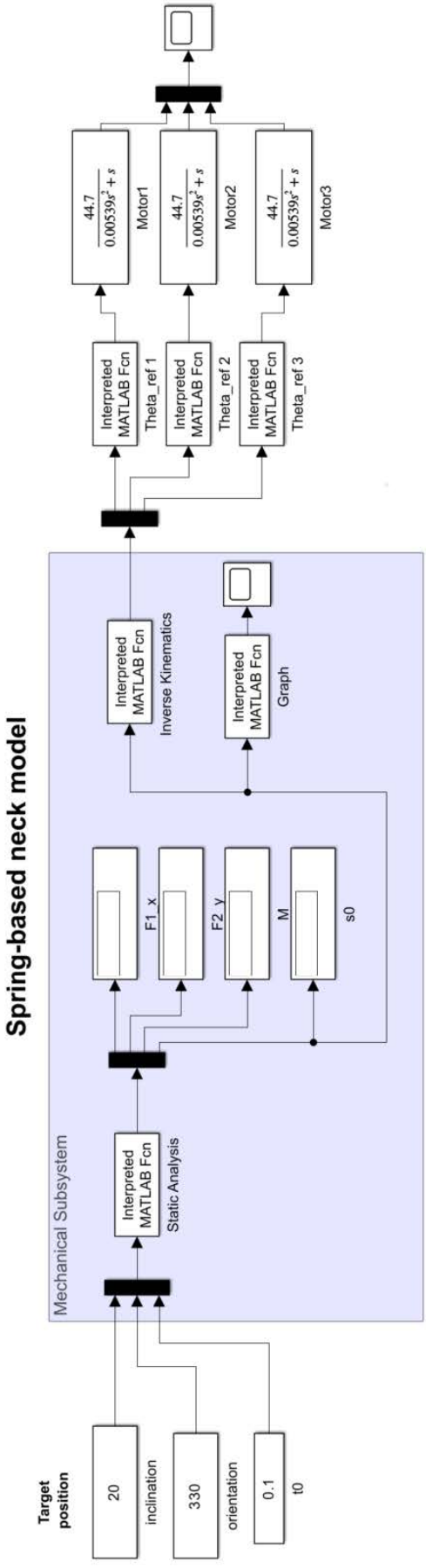


Figure 4.9: Simulink model of the spring-based robotic neck.

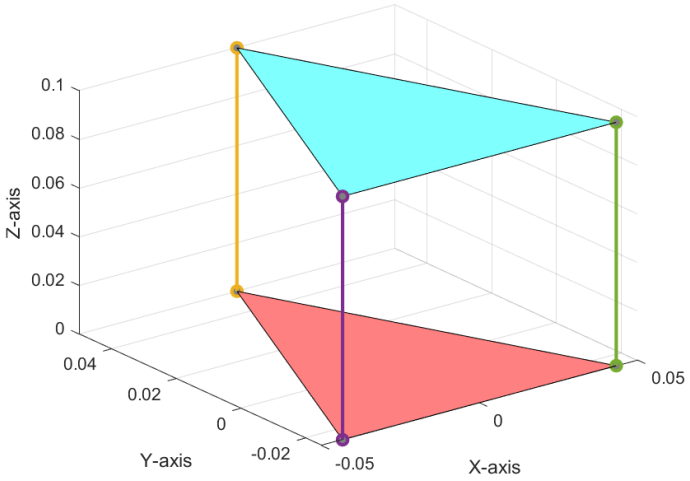
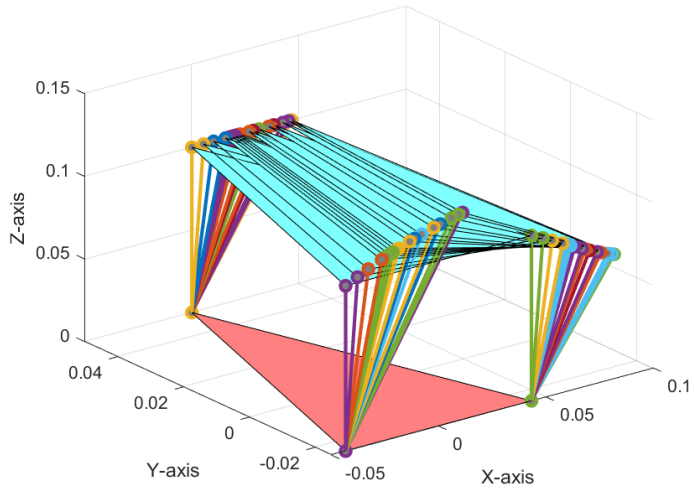
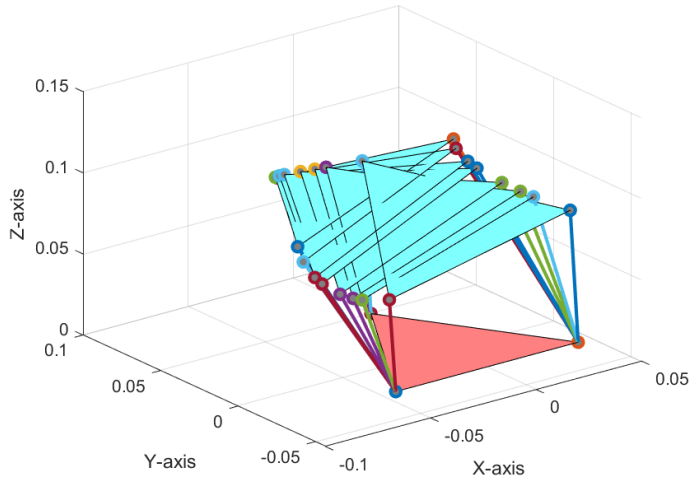


Figure 4.10: Simulated neck mechanism.



(a) Neck inclination.



(b) Neck orientation.

Figure 4.11: Simulation of movements of the neck.

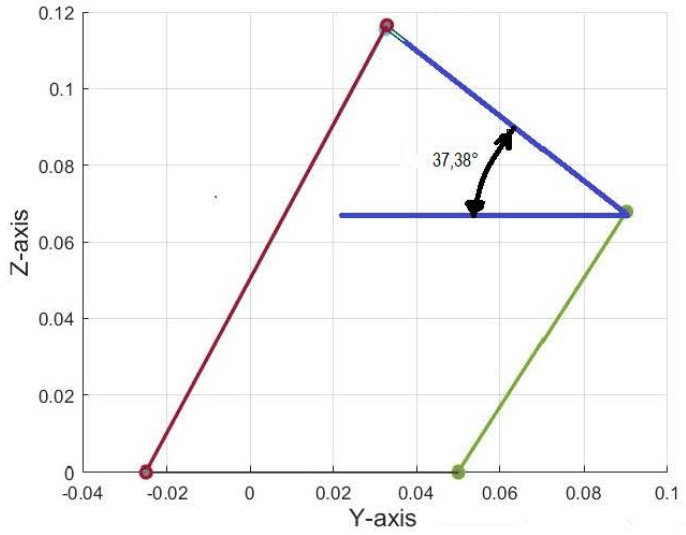


Figure 4.12: Theoretical maximum angle of inclination (40°).

$$L = f(x) \qquad f : \mathbb{R}^2 \longrightarrow \mathbb{R}^3 \tag{4.21}$$

In comparison with the spring-based neck, now the input workspace is $\mathbb{R}^2$, instead of $\mathbb{R}^3$. Equation (4.11) can be used to determine the cable lengths, where:

- ${}^{O}T_{o'}$ is the homogeneous transformation matrix that represents the projection from *oxyz* (mobile frame) to *OXYZ* (fixed frame):

$${}^{O}T_{o'} = \begin{bmatrix} {}^{O}R_{o'} & P_o \\ 0 & 1 \end{bmatrix}$$

where:

- P_o is the position vector of point o with respect to the base coordinate system:

$$P_o = \begin{bmatrix} s_o \cos\alpha & s_o \sin\alpha & t_0 \end{bmatrix}^T$$

- ${}^{O}R_{o'}$ is the rotational matrix that describes the orientation of the mobile platform using the Euler angles with *ZYZ* orientation:

$${}^{O}R_{o'} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

- $\overrightarrow{OA_i}$ represents the points of the fixed base (see Equation 4.1).
- $\overrightarrow{oB_i}$ represents the points of the mobile platform (see Equation 4.2).

4.2.1.1 Geometry configuration

As addressed in Section 4.1.4.1, the equilibrium of forces and moments of the spring-based neck was used to obtain the parameters s_0 and t_0, since the mechanical properties of the spring had to be taken into account. Now, when performing this new mathematical model, the soft link can be considered as a continuum with constant curvature (CC) [86]. This implies that s_0 and t_0 can be solved geometrically (see Figure 4.4), as follows:

$$R = \frac{L_o}{\alpha} \tag{4.22}$$

where L_o is the length of the soft link. Based on triangular geometries, the following equation is obtained:

$$R = s_o + R\cos\alpha \tag{4.23}$$

Therefore, the following relationship can be easily obtained from Equations (4.22) and (4.23):

$$s_o = \frac{L_o(1-\cos\alpha)}{\alpha} \tag{4.24}$$

In the same way, t_0 is obtained by a geometric relation:

$$t_o = \frac{L_o\sin\alpha}{\alpha} \tag{4.25}$$

whose constants values are given in Table 4.1.

Table 4.1: Soft link parameters and fixed and mobile base dimensions.

L_o[m]	a[m]	b[m]
0.107	0.05	0.05

Once parameters s_0 and t_0 have been calculated, the inverse kinematics equation (4.11) can be solved to obtain the values of the cable lengths. These values can be used to determine the target position of the soft neck platform.

The lengths L_i change in turn by varying the angular position θ_i of each of the three motors, whose relationship is determined by the radius r of the pulley attached to each motor:

$$\theta_i = \frac{L_o - L_i}{r} \tag{4.26}$$

4.2.2 Direct kinematics

For the direct kinematics analysis, Equation (4.12), which corresponds to the equilibrium of forces, will be used. This equation will help us to determine the direction of the mechanism for a given cable length. In addition, the dimensions of the fixed base are used to determine the limits of movement. With this information, the appropriate direction for the cable-driven mechanism can be defined to obtain the target position.

4.2.2.1 Force equilibrium equation

Equation (4.12) can be decomposed into three equations:

$$\begin{aligned} t'_1\ (R_{12}\,b + s_0\,\cos(\beta)) - t'_2\left(\frac{R_{12}\,b}{2} - \frac{\sqrt{3}\,a}{2} - s_0\,\cos(\beta) + \frac{\sqrt{3}\,R_{11}\,b}{2}\right) \\ -t'_3\left(\frac{R_{12}\,b}{2} + \frac{\sqrt{3}\,a}{2} - s_0\,\cos(\beta) - \frac{\sqrt{3}\,R_{11}\,b}{2}\right) - F_1\,\cos(\beta) \;=\; 0 \end{aligned} \tag{4.27}$$

$$\begin{aligned} t'_2\left(\frac{a}{2} - \frac{R_{22}\,b}{2} + s_0\,\sin(\beta) - \frac{\sqrt{3}\,R_{21}\,b}{2}\right) + t'_3\left(\frac{a}{2} - \frac{R_{22}\,b}{2} + s_0\,\sin(\beta) + \frac{\sqrt{3}\,R_{21}\,b}{2}\right) \\ -F_1\,\sin(\beta) + t'_1\ (R_{22}\,b - a + s_0\,\sin(\beta)) \;=\; 0 \end{aligned} \tag{4.28}$$

$$\begin{aligned} F_2 + t'_3\left(t_0 - \frac{R_{32}\,b}{2} + \frac{\sqrt{3}\,R_{31}\,b}{2}\right) - g\,m - t'_2\left(\frac{R_{32}\,b}{2} - t_0 + \frac{\sqrt{3}\,R_{31}\,b}{2}\right) \\ +t'_1\ (t_0 + R_{32}\,b) \;=\; 0 \end{aligned} \tag{4.29}$$

where ${}^{O}T_i = t'_i = t_i u_i$.

From (4.27) $\cdot \sin(\beta)$ - (4.28) $\cdot \cos(\beta)$, one has:

$$(a-b)\left(t'_2\,\cos(\beta) - 2\,t'_1\,\cos(\beta) + t'_3\,\cos(\beta) - \sqrt{3}\,t'_2\,\sin(\beta) + \sqrt{3}\,t'_3\,\sin(\beta)\right) = 0 \tag{4.30}$$

From Equation (4.30), we have:

$$\beta = \tan^{-1}\left(\frac{t'_2 - 2\,t'_1 - t'_3}{\sqrt{3}\,\left(t'_2 - t'_3\right)}\right) \tag{4.31}$$

We know that:

$$u_i = \frac{{}^{O}T_{o'}\overrightarrow{oB_i} - \overrightarrow{OA_i}}{\left\|{}^{O}T_{o'}\overrightarrow{oB_i} - \overrightarrow{OA_i}\right\|} = \frac{\overrightarrow{L_i}}{L_i}$$

Therefore, Equation (4.31) can be expressed as:

$$\beta = \tan^{-1}\left(\frac{L_2 - 2\,L_1 - L_3}{\sqrt{3}\,(L_2 - L_3)}\right) \tag{4.32}$$

Equation 4.32 yields the orientation angle β as a function of the lengths of the wires L_1, L_2 and L_3. The following section will address the calculation of the inclination angle α as a function of these lengths to completely solve the direct kinematics problem.

4.2.2.2 Geometry configuration

For the calculation of the inclination angle α, the geometry configuration of the neck is addressed, considering that the soft link bends with an almost constant curvature (CC approximation), as shown in Figure 4.4. The purpose is to find a complete relationship between the cable lengths and this inclination angle.

Figure 4.1 provides a view of the fixed base and the mobile platform of the soft neck of the robot, while Figure 4.13 shows a view of the fixed base of the mechanism from above when looking down along the z-axis. As Figure 4.4 shows, the radius R of uniform curvature measured from the center of the neck is related to the radius R_i of curvature of each cable (L_2, L_1, L_3):

$$R_i = R - a\,\cos\beta_i \quad (i = 1, 2, 3) \tag{4.33}$$

where a is the distance from the center of the neck to one of the cables and β_i is the angle between the soft link and cable i position.

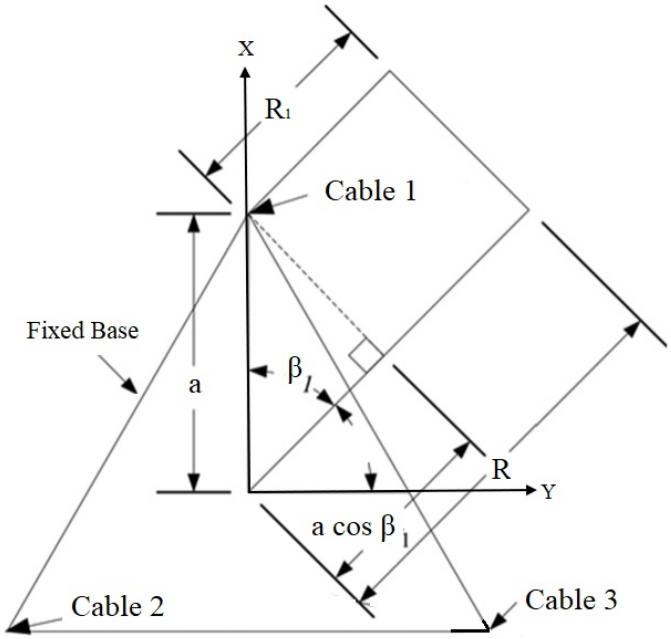


Figure 4.13: Diagram of the fixed base.

Let us consider that the soft link bends as a CC continuum with radius of curvature:

$$R = \frac{L_o}{\alpha} \tag{4.34}$$

where L_o is the length of the soft link.

By replacing Equation (4.34) into Equation (4.33), the relationship between the arc length of the robot L_o and the imaginary radius of the arc (L_{Oi}) is:

$$R_i = \frac{L_o}{\alpha} - a\cos\beta_i \quad (i = 1, 2, 3) \tag{4.35}$$

Recalling that the length L_i of a chord formed with the extreme points of the arc L_{oi} is given by $L_i = 2R_i \sin\frac{\alpha}{2}$, therefore, the i^{th} equation of the cable length is:

$$L_i = 2\sin\frac{\alpha}{2}\left(\frac{L_o}{\alpha} - a\cos\beta_i\right) \quad (i = 1, 2, 3) \tag{4.36}$$

The three cables are assumed to be equidistant around the soft link. Cables are separated by angles of 120^o (Figure 4.13) and are related to the plane Ost of the soft robotic neck by $\beta_1 = 90^o - \beta$, $\beta_2 = 210^o - \beta$ and $\beta_3 = 330^o - \beta$. Then, the cable lengths are defined by the following equations:

$$L_1 = \sin\left(\frac{\alpha}{2}\right)\left(\frac{2\,L_0}{\alpha} - 2\,a\cos\left(\beta - \frac{\pi}{2}\right)\right) \tag{4.37}$$

$$L_2 = \sin\left(\frac{\alpha}{2}\right)\left(\frac{2\,L_0}{\alpha} - 2\,a\,\cos\left(\beta - \frac{7\,\pi}{6}\right)\right) \tag{4.38}$$

$$L_3 = \sin\left(\frac{\alpha}{2}\right)\left(\frac{2\,L_0}{\alpha} - 2\,a\,\cos\left(\beta - \frac{11\,\pi}{6}\right)\right) \tag{4.39}$$

Considering the term $(L_2 - L_3)$ from Equation (4.32), the subtraction of Equations (4.38) and (4.39) results as follows:

$$L_2 - L_3 = 2\,\sqrt{3}\,a\,\sin\left(\frac{\alpha}{2}\right)\cos\left(\beta\right) \tag{4.40}$$

from which:

$$\alpha = 2\,\sin^{-1}\left(\frac{L_2 - L_3}{2\,\sqrt{3}\,a\,\cos\left(\beta\right)}\right) \tag{4.41}$$

Therefore, Equations (4.31) and (4.41) describe the direct kinematics of the soft neck, relating the orientation and inclination angles with the cable lengths.

4.2.3 Simulation of the mathematical model

Similarly to the case of the spring-based neck, the inverse kinematics and static model is implemented in ©Matlab. Varying α from 0° to 40° and β from 0° to 360° gives the results shown in Figure 4.14, where the cable lengths are in the z coordinate.

Figure 4.15 shows the system implemented in ©Matlab Simulink. From left to right, we have: the target position block, the geometry configuration block where the output variables are t_0 and s_0 as described in Section 4.2.1, the inverse kinematics block, the simulation graph block (graphical representation of the mechanism according to the parameters introduced), DC motors blocks (each cable is connected to its motor and cable lengths are translated into motor angular positions).

Figure 4.16 shows a three-dimensional simulation of the CDPM, which reproduces the movements of the soft link-based robotic neck. This simulation allows the response of the system to certain pose input values to be viewed.

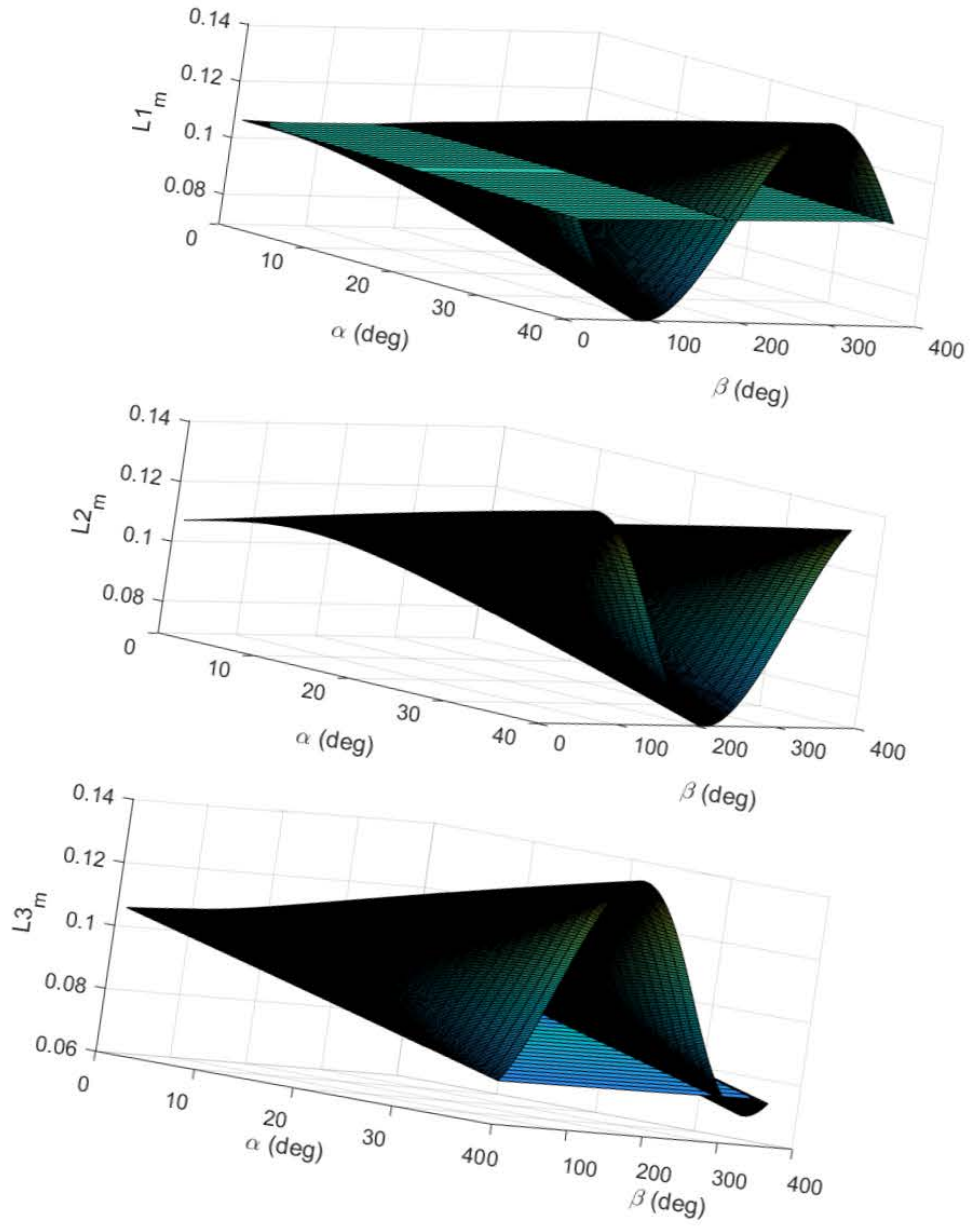


Figure 4.14: Inverse kinematics solution for the soft link-based neck.

4.3 Conclusions

The mathematical models for the cable-driven parallel mechanisms proposed in this thesis have been developed for both spring-based and soft link-based necks. The models have been validated through simulation in ©Matlab Simulink, where the cable lengths are calculated as a function of the desired neck pose. The results show that both models can be used for neck motion control applications. The soft link-based neck has the advantage of not requiring springs,

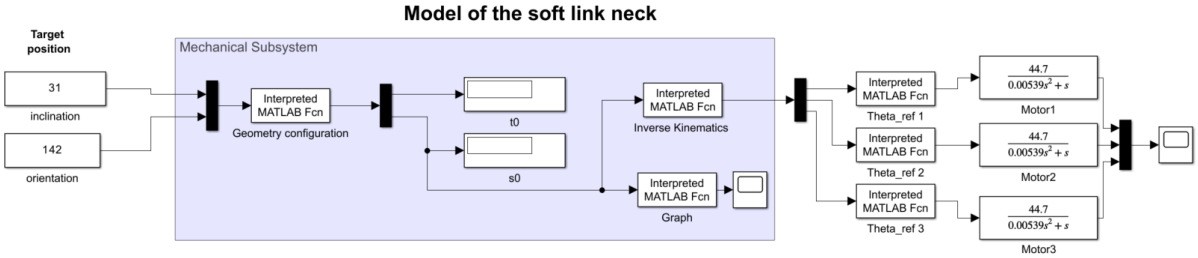


Figure 4.15: Simulink model of the soft link-based robotic neck.

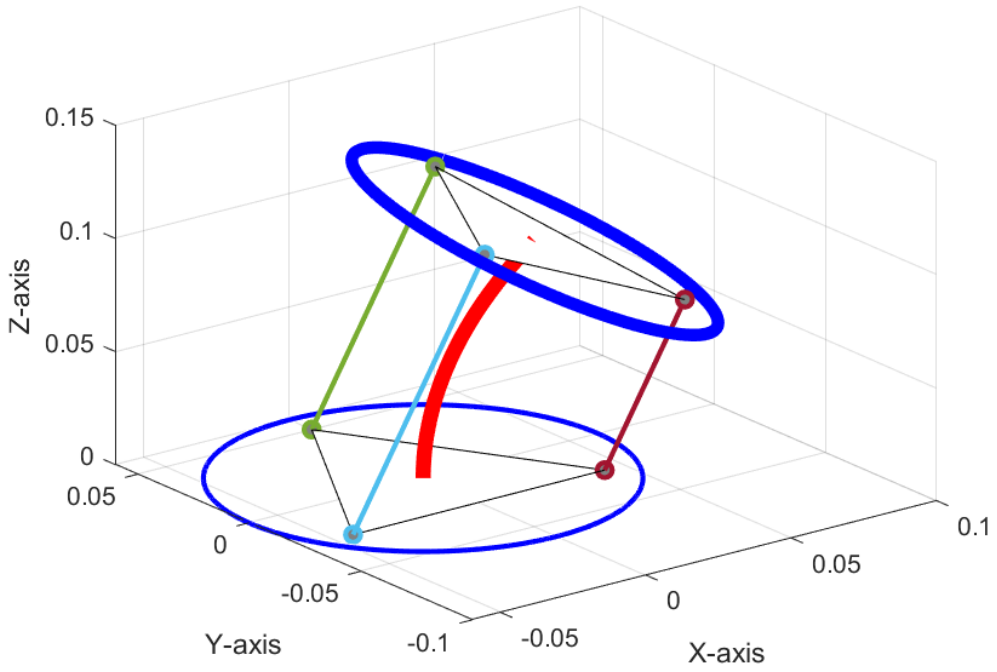


Figure 4.16: Simulated neck mechanism.

while the spring-based neck requires the inclusion of the mechanical properties of the spring into the model. The simulations demonstrate the responsiveness of the systems to postural inputs, which will be later validated through experimental tests on the different platforms.

Chapter 5

MACHINE LEARNING-BASED SOFT JOINT MODELING

This chapter presents a machine learning approach to the problem of identifying the inverse kinematics of the soft robotic neck. A multilayer perceptron is used to build an identification system that obtains a relationship between the desired neck poses and the angular configuration of the three motors that control the neck. The performance of the proposed approach is validated by simulation and experimental results.

5.1 Introduction

According to [87], robot control requires accurate models to achieve a high level of robot performance, otherwise the robot and its environment are endangered. On the other hand, due to the complexity of modern robotic systems, standard analytical models are approximations that do not accurately model the nonlinearities of such systems, which come from, for example, their inherent actuation systems, from the mechanical friction of their structure or other sources of friction, or from the deformation of certain components of the system, such as cables or links.

For this reason, data-based model learning methods have become increasingly interesting tools for robot modeling. For example, neural networks are widely used to identify inverse kinematics for both redundant [88]–[90] and non-redundant [91], [92] manipulator robots.

The problem of obtaining inverse kinematics is particularly severe in the case of soft robots, whose model is highly dependent on the characteristics of each robot, its morphology and the type of soft material it is manufactured from. For example, in [93] a machine learning based

approach for the development of dynamic models for a soft robotic manipulator and a trajectory optimisation method for predictive control of the manipulator in the task space are addressed. In [94] an artificial neural network, more specifically a feed-forward multilayer perceptron (MLP), is trained for use in calculating the inverse kinematics of a robotic manipulator. And in [95] an optimisation method is proposed that solves the inverse kinematics of redundant robots based on the damped least squares (DLS) method. In each case, the kinematics of the robots are very different and the proposed solution is very focused on the specific problem of each robot.

Section 4.2 addresses the inverse kinematics model of the soft link-based robotic neck proposed in this thesis. As previously described, the analytical model obtained is based on the assumption of a constant curvature characteristic of the neck when the actuating forces that allow its movement are applied, and on ignoring the influence of the elasticity of the soft material that makes it up. These assumptions reduce the accuracy of the resulting model, which does not adequately represent the behaviour of the robot in certain areas of its working field. Although these mismatches between the theoretical model and the experimental model can be dealt with from a robust control perspective [96], [97], it is advisable to consider other more efficient modeling techniques for this type of system, in particular data-based techniques, as discussed below.

5.2 Machine learning-based models

Machine learning is a field of research at the intersection of statistics, artificial intelligence and computer science. It is also referred to as the process of predictive analytics or mathematical learning. According to [98], this learning is the design and study of software artefacts that use past experience to make future decisions; it is the study of programs that learn from data. Its basic goal is to induce an unknown rule from examples of its application. Supervised learning is used in this work, as the algorithms used are trained on a database where the inputs and outputs are already defined, as described in Section 5.3.

5.2.1 Artificial Neural Network - Multilayer Perceptron

One of the most common models used in machine learning is the model based on artificial neural networks, or simply neural networks. According to [99], an artificial neural network is a type of machine learning model inspired by the structure and function of biological neural

networks. It consists of layers of interconnected nodes, called artificial neurons, which are activated or deactivated according to a set of mathematical rules (activation functions). Neural networks are characterised by the type of activation function used to produce an output and by the architecture of the network in terms of how the nodes are connected. According to [100], a Multilayer Perceptron (MLP) is a neural network that has different layers of neurons. The input layer consists of the inputs to the neural network and the output layer consists of the final outputs of the MLP. A hidden layer is any intermediate layer, and an MLP can have one or more hidden layers.

Two of the most common supervised machine learning tasks are classification and regression. In supervised learning, classification tasks require these models to learn to predict discrete values for "response" variables from one or more explanatory variables. That is, for new observations, the program must predict the most likely category, class or label. On the other hand, in regression problems, the model must predict the value of a continuous variable. This means that the solution to this problem is represented by a continuous variable that can be flexibly determined by the inputs to our model, rather than being restricted to a set of possible values determined by classes, as in the case of classification. Basically, this study focuses on the regression task, the objective of which is to predict the values of the angular position of the three motors acting on the neck (values in radians) in order to achieve the given values of inclination and orientation.

Next, the effectiveness of using neural networks to model the soft robotic neck described in Section 4.2.1 is evaluated. The influence of the number of layers in the network and the different activation functions chosen will be investigated and described in detail in the following section.

5.2.2 Activation functions

Activation functions, according to [101], are used specifically in artificial neural networks to transform an input signal into an output signal, which in turn serves as an input to the next layer of neurons. In an artificial neural network, the sum of the products of the inputs and their corresponding weights is calculated and finally an activation function is applied to obtain the output of that particular layer, which is fed as input to the next layer.

A number of factors need to be taken into account to achieve the best possible result with the least possible error, such as the number of hidden layers in a network, the training methods, the setting of the hyper-parameters and the activation function. The latter is one of the most

important parameters to consider.

Choosing the appropriate activation function for a particular system is a difficult process, as it depends on the case of the data and its application. Also, the choice of activation function in the output layer will have more weight than the one used in the hidden layers. A hidden layer in a neural network is a layer that receives inputs from another layer (such as another hidden layer or an input layer) and provides outputs to another layer (such as another hidden layer or an output layer). A hidden layer does not directly touch input data, nor does it produce outputs for a model, at least not in general. According to [102], a nonlinear differentiable activation function is typically used in the hidden layers of a neural network. This allows the model to learn more complex functions than a network trained with a linear activation function. In the case of the soft neck, which has both positive and negative data (clockwise and counterclockwise rotations) in the angular positions of the three motors, activation functions whose values include positive and negative values are used to achieve optimal results in the predictions. Therefore, we will use the Tangent Hyperbolic (TANH) and Exponential Linear Unit (ELU) activation functions, which are the most commonly used in the hidden layers of neural networks. These are defined below, together with the sigmoid activation function.

5.2.2.1 Tangent Hyperbolic (TANH) activation function

The hyperbolic tangent defined in Equation (5.1) is similar to the sigmoid function (another activation function described below). However, this one is symmetric about the origin. This gives different signs to the outputs of previous layers, which are fed as inputs to the next layer, as described in [103].

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{5.1}$$

The TANH function (Figure 5.1) is continuous and differentiable and its values lie between -1 and 1. Compared to the sigmoid function, the gradient of the TANH function is steeper. It is preferable to use this function over the sigmoid function because it has gradients that are not restricted to varying in a particular direction and it is also centred at 0. This means that the function takes any real values as input and output values in the range -1 to 1. The larger the input (more positive), the closer the output will be to 1, while the smaller the input (more negative), the closer the output will be to -1. Given the above, using this activation function will fit the data better as the angular positions of the motors are in the same range of values as the

activation function.

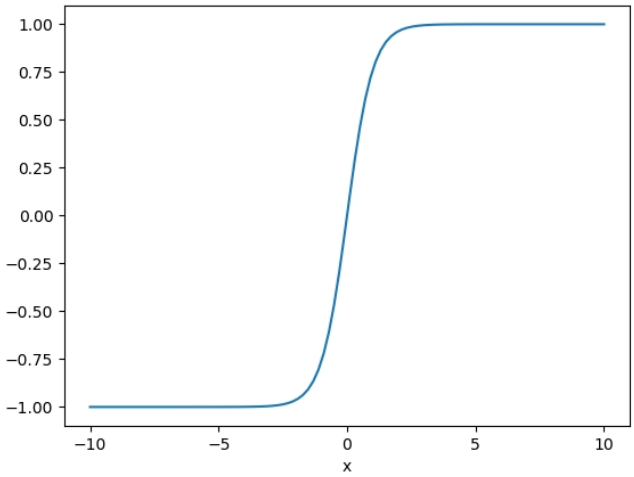


Figure 5.1: TANH function.

5.2.2.2 Exponential Linear Unit (ELU) activation function

The Exponential Linear Unit or ELU, defined in equation (5.2) (Figure 5.2), is a variant of the Rectified Linear Unit (RELU). Unlike RELUs, ELUs have negative values, which allows them to move mean unit activations closer to zero, like batch normalisation, but with less computational complexity. Mean shifts towards zero speed up learning by bringing the normal gradient closer to the unit's natural gradient due to a reduced bias shift effect [104]. This function introduces a slope parameter for negative x values. It uses a logarithmic curve to define the negative values, as described in [101]. This means that by using this activation function, optimal data prediction results can be obtained because the angular positions of the motors are in the same range of values as this function and tend to converge to zero faster, producing more accurate results.

$$\begin{aligned} f(x) &= x \quad (x \geq 0), \\ f(x) &= a(e^x - 1) \quad (x < 0). \end{aligned} \tag{5.2}$$

5.2.2.3 Logistic activation function

The logistic function or sigmoid activation function, defined in Equation (5.3), is a commonly used activation function in neural networks. This function is continuous and differentiable, and its values lie between 0 and 1 (Figure 5.3). The logistic function is a nonlinear

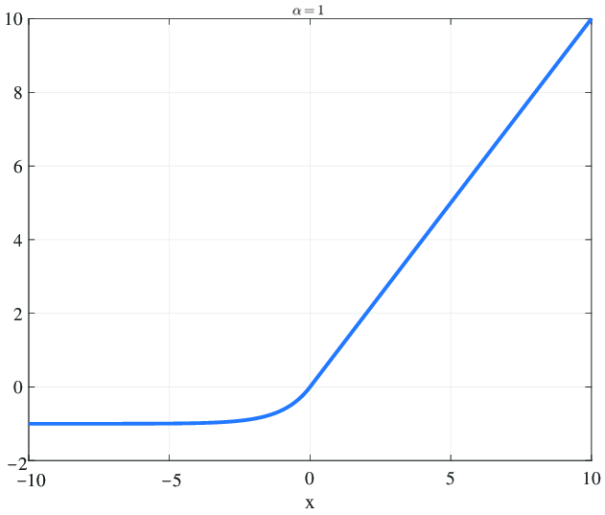


Figure 5.2: ELU function.

function and its gradient is limited to vary in one direction. This means that if the input is positive, the output will always be positive and vice versa. However, despite this limitation, the logistic function is a good choice to fit the data as the input and output values are normalized between 0 and 1. Therefore, using this function will ensure that the data is in the correct range for proper interpretation.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{5.3}$$

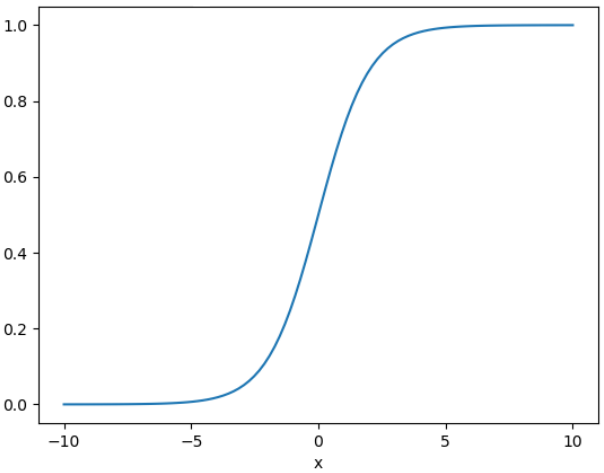


Figure 5.3: Logistic function.

5.3 Simulation setup

This section describes the data used to train the different MLPs.

5.3.1 Data extraction

The following datasets are used to identify the soft robotic neck model:

- **Pose data:** a 3DM-GX5-10 Inertial Measurement Unit (IMU) sensor is used to capture the inclination and orientation (poses) of the neck in real time.

- **Sampling time:** During data acquisition, the sensor requires a settling time of approximately two seconds to accurately measure neck inclination and orientation, as shown in Figure 5.4. Therefore, this value is set as the sampling time during data collection.

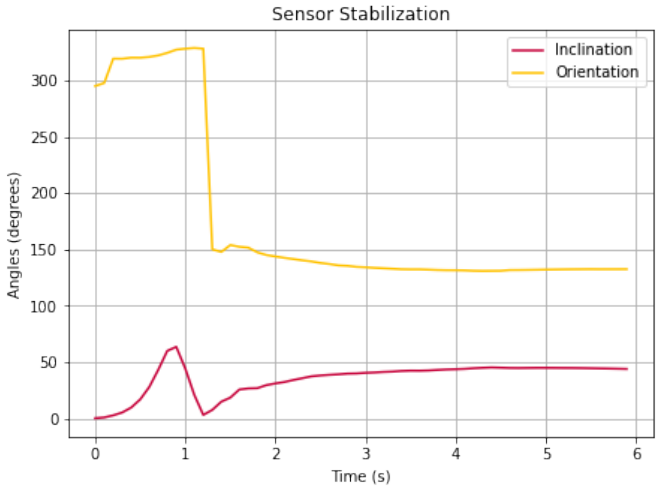


Figure 5.4: Sensor stabilization time.

- **Pose trajectory:** the planned trajectory for the data acquisition is as follows: the inclination varies from 5° to 35° in steps of 5°; for each of the inclinations, the orientation varies from 0° to 360° in steps of 10°. A total of 28,800 data points are obtained. For each inclination/orientation pair, a sampling time of two seconds is set due to the stabilisation of the sensor. In other words, we obtain 100 data for each pose in the sampling time. This trajectory is commanded on the real robot platform using the analytical inverse kinematics model described in Section 4.2.1 to calculate the reference angular positions corresponding to each of the three motors.

- **Encoder data:** The actual angular positions achieved by each of the three motors are measured by their respective encoders (position sensors) in radians and are shown in Figure 5.5.

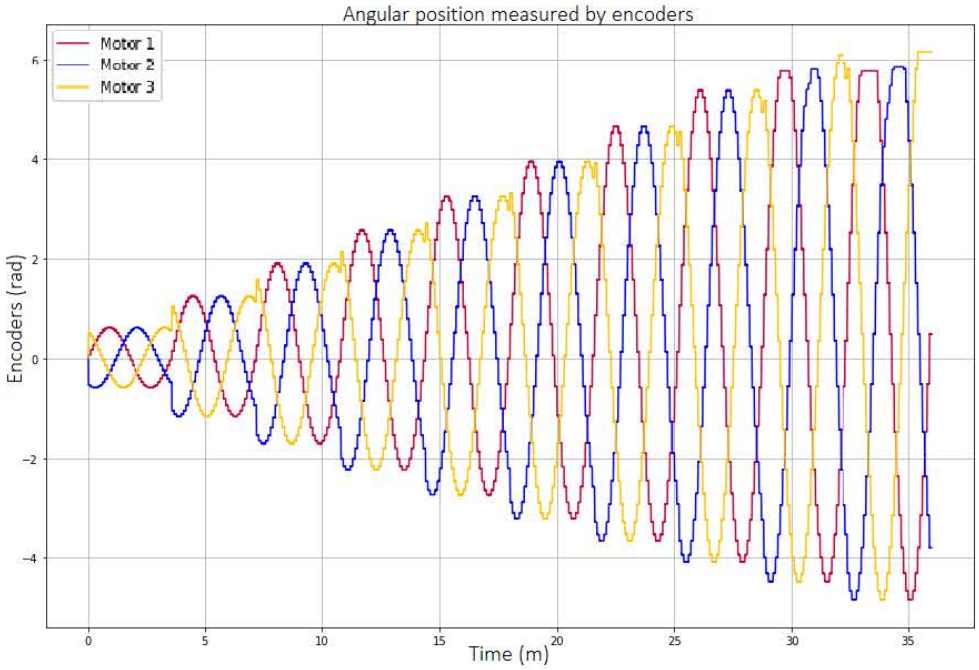


Figure 5.5: Motor angular positions measured by the encoders (in radians).

5.3.2 Training of the models

This study deals with the regression task, the objective of which is to predict the values of the angular positions of the three motors (which are continuous values) for the given values of inclination and orientation (in degrees). The inputs and outputs of the network are defined as follows:

- **Input:** Inclination and orientation values extracted from the IMU sensor, given the motion produced by the angular positions of the motors calculated according to the inverse kinematics analytical model of the soft robotic neck.
- **Output:** The angular positions of the three motors as measured by the motor encoders.
- **Data partitioning:** Most machine learning modeling processes use a 70/30 or 80/20 ratio between training and testing data sets, according to [105], [106]. In our case, 70% of the data is used for training and 30% for validation.

5.4 Simulation results

In this section, a simple MLP, a modified MLP (different layer architectures) and an optimized MLP (based on hyper-parameters) are trained and compared in simulation.

5.4.1 Simple MLP

The simple MLP is trained with 3 hidden layers of 5 neurons each, as shown in Table 5.1.

Table 5.1: MLP Parameters [105].

Parameters	**Values**	**Description**
Hidden layers sizes	(5,5,5)	Number of hidden layers and number of neurons in each layer.
Activation function	logistic	It refers to how the weighted sum of the input is transformed into an output from a node in a layer.

After the MLP training process, the predicted motor angular positions for the three motors commanding the soft robotic neck are obtained and represented in Figure 5.6. The graph formed as a result of these predicted data shows a strong similarity with that in Figure 5.5, which was used as input for the training.

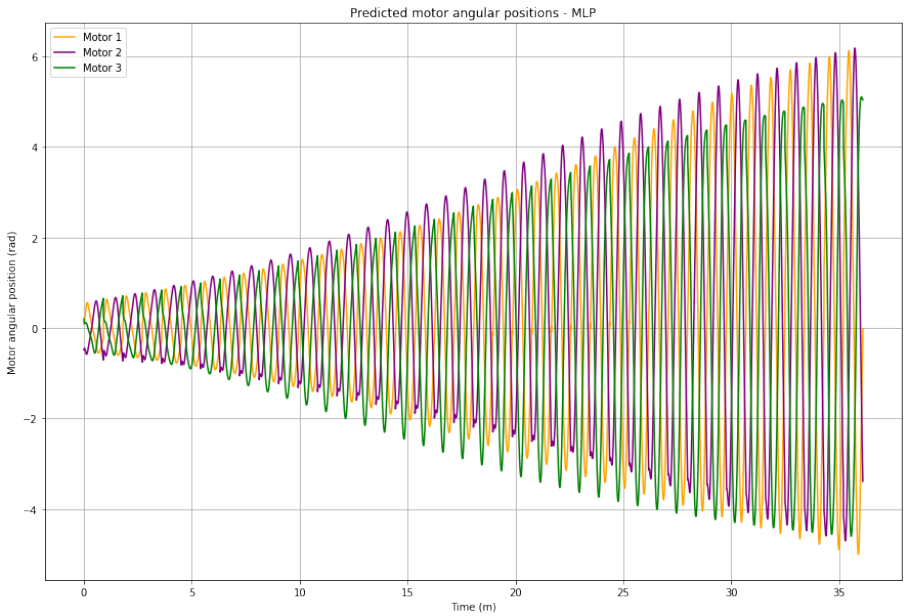


Figure 5.6: Predicted motor angular positions using a simple MLP.

5.4.2 Modified MLP

Now, the MLP approach is modified taking into account different architectures for the hidden layers, as described next.

5.4.2.1 Architectures selection

Simulations are performed to evaluate the behaviour of the neural network consolidation. Three different architectures are used to analyse which would be the optimal architecture to achieve the best results.

- **Symmetric:** The first layer will have five neurons and their number will increase by five in each layer until half of the hidden layers are reached, and then it will decrease by five and by five, as shown in Figure 5.7.

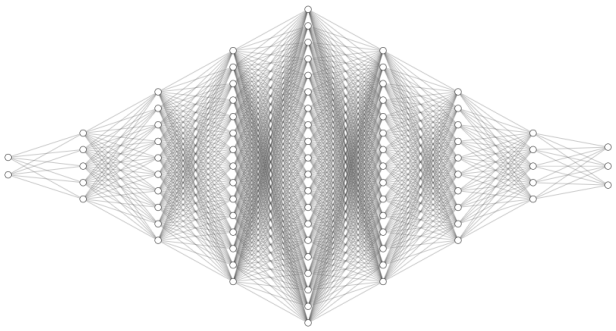


Figure 5.7: Symmetric architecture.

- **Constant:** As shown in Figure 5.8, in this architecture all hidden layers will have 25 neurons.

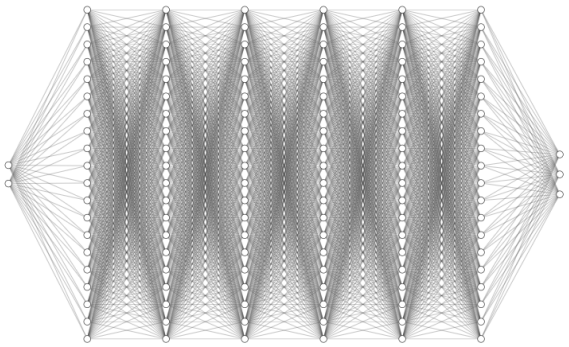


Figure 5.8: Constant architecture.

- **Alternating:** For this architecture, the hidden layers will have 10 and 20 neurons in alternation, as shown in Figure 5.9.

Figure 5.10 shows the nine models obtained using the three types of architectures presented, varying only the number of layers (10, 20 and 30) for each of them and selecting the

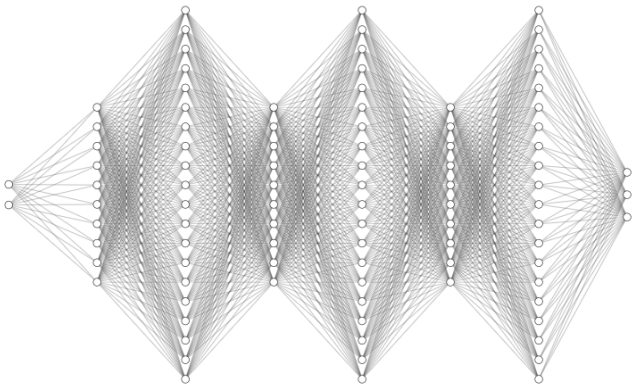


Figure 5.9: Alternating architecture.

TANH activation function. All the models presented are trained with 100 epochs (full cycle of trained data) in order to minimise the difference between the real and the theoretical results.

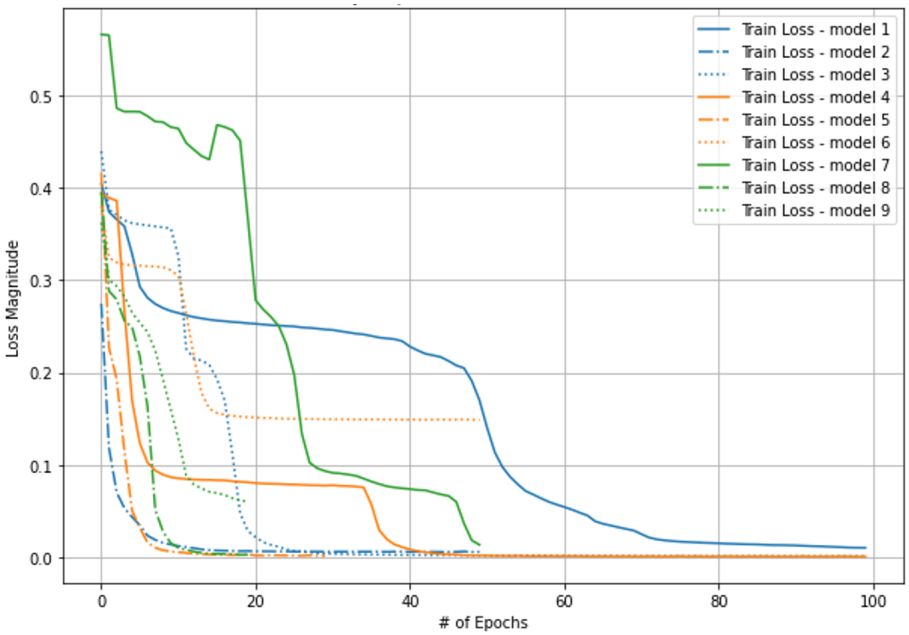


Figure 5.10: Magnitude of loss using the TANH activation function.

Figure 5.11 shows the nine models obtained under the same conditions as in the previous case, but selecting the ELU activation function.

The results obtained show different aspects. For example, the architectures have considerable adaptability with the ELU activation function, since it needs fewer epochs to obtain an acceptable loss magnitude, while for the TANH function more epochs are needed to obtain the same accuracy. In other words, we can say that in the architectures where the TANH function is applied, a greater effort is made when trying to generalise the prediction. It is also observed

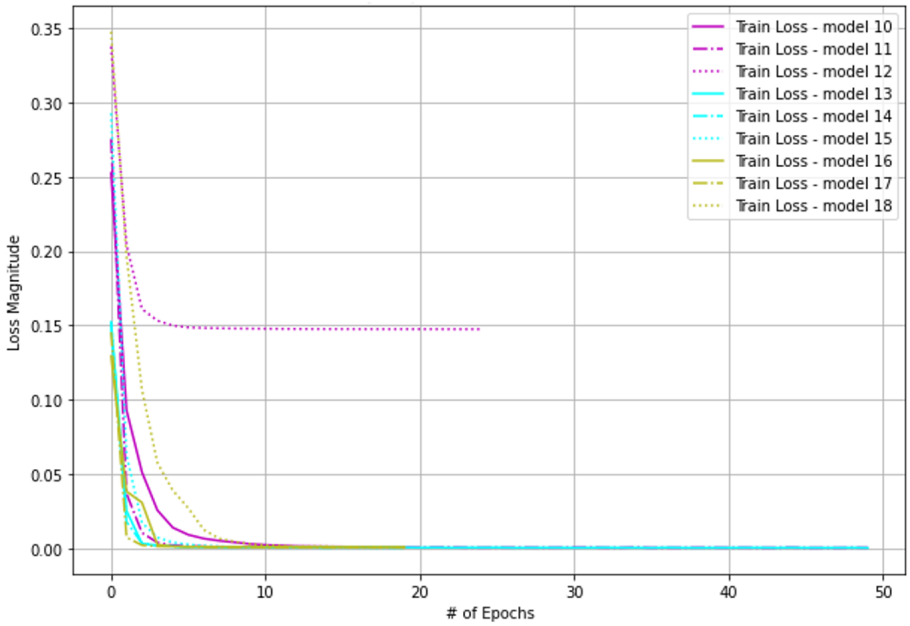


Figure 5.11: Magnitude of loss using the ELU activation function.

that the higher the number of hidden layers, the lower the number of epochs required. Finally, the constant architecture model performs better compared to the other two models.

Following these simulation results and the conclusions reached, two different neural networks are selected for the characterisation of the kinematic model of the soft neck, whose characteristics are defined in Table 5.2.

Table 5.2: Selected Neural Networks.

Architecture	**# of Layers**	**# of Neurons**	**Activation Function**
Constant	10	25	TanH
Constant	10	25	Elu

5.4.3 Multilayer Perceptron - Hyper-parameters Optimization

Now that a simple and modified MLP have been proposed and discussed, another MLP approach based on hyper-parameters optimization is addressed.

In general, it is good practice to use multiple hidden layers, as well as multiple nodes within the hidden layers, as this seems to result in better performance. According to the work of Ian Goodfellow (the creator of the generative adversarial network) [107], increasing the number of layers of neural networks tends to improve the overall accuracy of the test set. To explore how these parameters affect the training process, a hyper-parameters optimisation was performed

using the GridSearchCV function of the Scikit-learn library [105], which allows us to analyse the performance for different parameter combinations, as described in Table 5.3.

Table 5.3: Parameter Optimization [105].

Parameters	**Values**	**Description**
Hidden layers sizes	(5,5), (10,10), (100), (50,50,50), (50,100,50)	Number of hidden layers and number of neurons in each layer.
Activation function	logistic	It refers to how the weighted sum of the input is transformed into an output from a node in a layer.
Solver	sgd, adam	- 'sgd' refers to stochastic gradient descent. - 'adam' refers to a stochastic gradient-based optimizer.
Alpha	0.0001, 0.05	L2 penalty (regularization term) parameter.
Learning rate	constant, adaptive	- 'constant' refers to a constant learning rate. - 'adaptive' keeps the learning rate constant to 'learning_rate_init' as long as the training loss keeps decreasing.

After using this parameter optimisation function with the variables shown in Table 5.3 to identify the soft robot neck, the neural network with 3 hidden layers of 50 neurons each and the parameters shown in Table 5.4 is found to be the best performing.

Table 5.4: Best Performer from the hyper-parameters optimization.

Hidden layers sizes	**Act. function**	**Solver**	**Alpha**	**Learning rate**
(50,50,50)	logistic	adam	0.0001	constant

After the simulation of this optimized MLP model, Figure 5.12 presents the predicted motor angular positions for the pose trajectory defined, showing a slimmer cone shape in comparison with the previous MLP approaches.

The MLP models presented in this section will be validated through real experiments on the soft robotic neck in Section 6.6.

5.5 Conclusions

A machine learning approach to identify the inverse kinematics of the soft link-based robotic neck has been presented in this chapter. A multilayer perceptron has been used to build an identification system that obtains the relationship between the desired neck poses and the

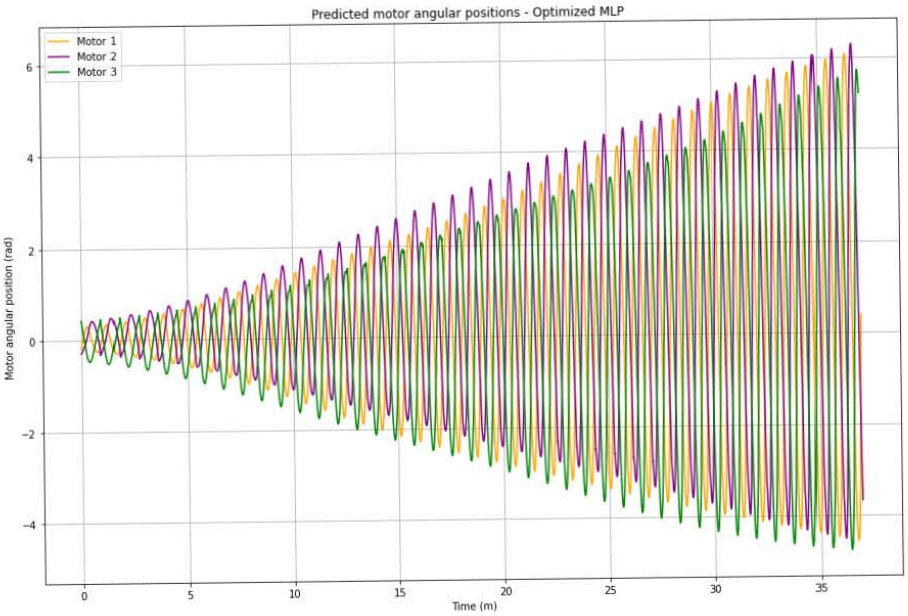


Figure 5.12: Predicted motor angular positions using optimized MLP.

angular configuration of the three motors that control the neck. The influence of the different activation functions and architectures on the performance of the MLP has been analysed through simulation results. In addition, a hyper-parameter optimisation approach has been presented. The results demonstrate the suitability of neural networks to model the complex behaviour of soft robots. The proposed MLP models will be validated through real experiments in the next chapter.

Chapter 6

EXPERIMENTAL RESULTS

In this chapter the designs and models of the different soft necks proposed in this thesis are tested and validated. Different tests are performed to evaluate their performance and their final integration into the humanoid robot TEO.

6.1 Control problem

Before testing the performance of the soft robotic platforms, the motor control problem must be first addressed at low level. This section describes the control loop implemented for the inner control of the three motors commanding the tendons of the soft necks. In order to reach and hold the desired motor angular positions (θ_i) corresponding to each desired neck pose (α, β), the position control loop described in Figure. 6.1 is proposed for each motor, where the position error (θ_e) is obtained from the encoder data (θ_m) and computed through a Fractional Order Controller (FOC), resulting in a velocity control action (ω_r) that commands the motor speed (ω).

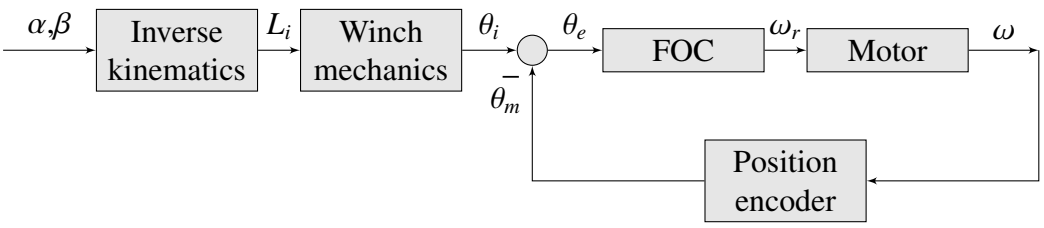


Figure 6.1: Soft neck control scheme for a single motor, showing: angular position error (θ_e), Fractional Order Controller (FOC), motor, angular speed (ω), position encoder, and angular position (θ_m).

In this control scheme, the motor plus encoder system is commanded by a velocity input

and provides a position output. The resulting system model obtained from identification [97] is given by the following transfer function:

$$G(s) = \frac{54.89}{54.89 + s} \cdot \frac{k_{enc}}{s}, \tag{6.1}$$

corresponding to a first order motor velocity model connected to an integrator (encoder model), where $k_{enc} = 6$ is the encoder gain that converts the rpm input into deg/s (360/60). The Bode diagram of the motor system in open loop and its corresponding time response in closed loop without controller are shown in Figure 6.2.

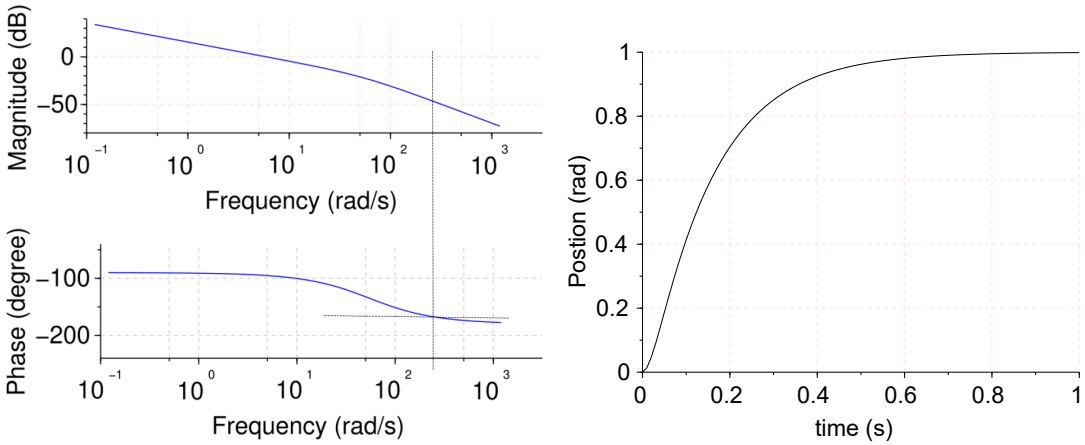


Figure 6.2: Motor model frequency response (Bode diagram of $G(s)$) (left) and time unitary feedback response (right).

Fractional order control [108] has demonstrated to be a robust control technique that enhances the potential of integer order proportional integral derivative (PID) controllers thanks to the introduction of the fractional (non-integer) orders in their derivative and integral actions, which increases the number of controller tuning parameters and so the number of control specifications to achieve. The general transfer function of a fractional order PID controller is given by

$$C(s) = K_p + K_i s^{-io} + k_d s^{do}, \tag{6.2}$$

where k_p is the proportional gain, k_i is the integral gain, k_d is the derivative gain and io and do are the fractional integral and derivative orders, respectively.

It is not the aim of this thesis to address in detail the control problem using FOC, since this has been deeply addressed in a previous thesis by Doctor Jorge Muñoz [109]. This section simply aims to describe the FOC used here for the inner motor control loop. A deeper analysis

can be found in [97].

The specific controller proposed in this case is a fractional order proportional integral (FOPI) controller, a generalization of the proportional integral (PI) controller where the integral action can have any fractional order (*io*):

$$C(s) = K_p + K_i s^{-io}. \tag{6.3}$$

The controller is tuned using Monje's tuning method described in [97] for the fulfillment of control frequency specifications such as gain crossover frequency ω_{cg} (directly related to the speed of the system), phase margin ϕ_m (directly related to the damping/overshoot of the response) and robustness to model gain changes (constant overshoot of the response for variations of the model gains, the so called iso-damping property). In order to obtain a fast and robust performance, the selected control specifications are $\omega_{cg} = 12 rad/s$ and $\phi_m = 60°$, which provides the following controller:

$$C(s) = 1.7595866 + 4.7817906 s^{-0.8}, \tag{6.4}$$

with proportional gain $k_p = 1.7807044$, integral gain $k_i = 4.8890371$ and integral fractional order $io = -0.81$. The corresponding Bode diagram of the open loop system with the FOC controller is shown in Figure 6.3 (left). It can be observed that the gain crossover frequency and phase margin specifications are fulfilled together with the characteristic of a flat phase curve around the gain crossover frequency, which allows a constant phase margin for model gain variations around the nominal one, fulfilling the iso-damping property of the time response of the system.

The time responses of the system in closed loop for different motor gains (G) are presented in Figure 6.3 (right), where a constant overshoot can be observed in all the responses thanks to the iso-damping property achieved by the FOC.

Now that the inner control loop has been closed for each motor with the help of the encoders in order to guarantee a robust motion performance of the neck, the need of integrating an inertial sensor into the system is addressed in the following section.

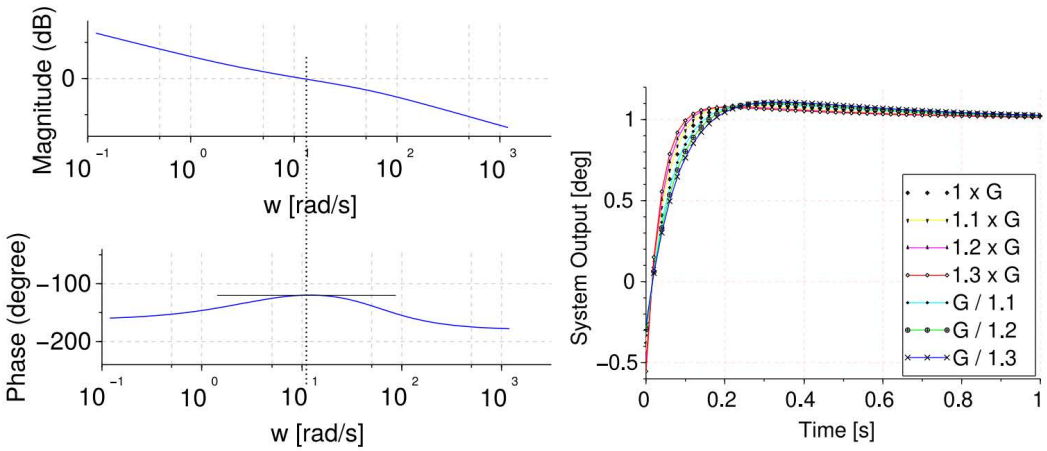


Figure 6.3: Open loop Bode diagram (left) and step response (right) of the system with the $FOPI$ controller based on Monje's method. $\phi_m = 60°$, $\omega_{cg} = 12 rad/s$.

6.2 Soft neck sensory system

In order to improve the sensory capabilities of the soft neck prototypes beyond encoder sensors, an Inertial Measurement Unit (IMU) sensor has been integrated into the system so that position data from the neck tip can be directly obtained and processed through an external control loop.

The low cost MPU-9250 IMU is chosen for this purpose. This 9-axis motion tracking device includes a 3-axis gyroscope, a 3-axis accelerometer, a 3-axis magnetometer and a Digital Motion Processor™ (DMP) in a small 3x3x1mm package. It is chosen for its reliability and capability to achieve the measurement speed required to measure the inclination and orientation angles of the neck.

6.2.1 Library

The MPU-9250 sensor implements a I^2C communication that allows the direct connection and communication with Arduino development boards, which will be used as data acquisition card in this case. Sparkfun markets the MPU-9250 sensor and they have developed an open source library in C++, compatible with the Arduino IDE. It consists of two libraries:

- Library MPU9250: Allows obtaining the data of the accelerometer, gyroscope and magnetometer in raw format.
- Library quaternionFilters: Calculates the corresponding sensor angles yaw, pitch and roll, through each of the components obtained by the MPU-9250.

The quaternionFilters library implements an orientation filter for matrices of inertial and magnetic sensors as proposed in Madgwick (2010), where acceleration, rotation speed and magnetic moments are merged to produce an estimate based on quaternions of absolute orientation of the device, which can be converted into roll, pitch and yaw angles. The performance of the filter is as good as conventional Kalman-based filters, with lower computational cost.

In this case, the Mahony filter has been used, which uses a similar scheme to Madgwick but employs a proportional and integral filtering in the error between the estimated and measured reference vectors [110].

6.2.2 Conversion of angles

The sensor library establishes an axes and rotation configuration as: the accelerometer axis X is aligned with the axis Y of the magnetometer, so the axis Z of the magnetometer is opposite in the direction of the axis Z of the accelerometer and the gyroscope.

The library generates the quaternions from which the yaw, pitch and roll angles are obtained. These are the Tait-Bryan angles, commonly used in aircraft orientation (see Figure 6.4).

In this coordinate system, the positive Z axis is down to the ground. Yaw is the angle between the sensor's X axis and the Earth's magnetic north. Pitch is the angle between the X axis of the sensor and the earth plane; towards earth it is positive and towards sky it is negative. Roll is the angle between the sensor's Y axis and the earth plane; the upward Y axis is positive roll.

These angles come from the definition of the homogeneous rotation matrix constructed by quaternions. Tait-Bryan angles, as well as Euler angles, are not commutative; that is, to get the correct orientation, the rotations must be applied in the correct order, which for this configuration is yaw, pitch and roll.

The Tait-Bryan matrix with the corresponding yaw (ψ), pitch (θ) and roll (ϕ) angles looks as follows:

$$R_{\phi,\psi,\theta} = \begin{bmatrix} C\psi C\theta & (C\psi S\theta S\phi - S\psi C\phi) & (C\psi S\theta C\phi + S\psi S\phi) \\ S\psi C\theta & (S\psi S\theta S\phi + C\psi C\phi) & (S\psi S\theta C\phi - C\psi S\phi) \\ -S\theta & C\theta S\phi & C\theta C\phi \end{bmatrix} \tag{6.5}$$

Only R_θ is considered because angle ψ corresponding to the *Yaw* angles is annulled for the neck platform, since it only has two DOF: one for inclination in *Pitch* and the other for

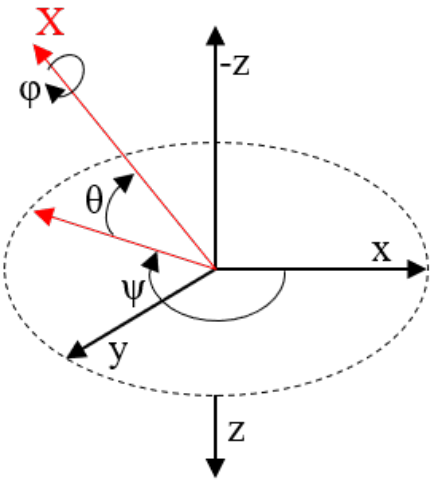


Figure 6.4: Tait-Bryan angles: yaw (ψ), pitch (θ) and roll (ϕ).

orientation in *Roll*. So the Tait-Bryan matrix is reduced to the vector:

$$R_{TB} = \begin{bmatrix} S\theta C\phi \\ -S\phi \\ C\theta C\phi \end{bmatrix} \tag{6.6}$$

The inclination angle (α) of the platform is the projection of the vector R_{TB} on the Z axis, according to the following equations:

$$cos\alpha = \frac{R_{TB}.Z}{||R_{TB}||.||Z||} = \frac{\begin{bmatrix} S\theta C\phi \\ -S\phi \\ C\theta C\phi \end{bmatrix} . \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}{||R_{TB}||.||Z||} \tag{6.7}$$

$$cos\alpha = \frac{C\theta C\phi}{\sqrt{(S\theta C\phi)^2 + (-S\phi)^2 + (C\theta C\phi)^2}} \tag{6.8}$$

The orientation angle (β) is obtained from the projection of the vector R_{TB} on the XY plane. In other words, there are two possible complementary solutions for orientation. In this case, we consider the solution of the X axis that does not need to add arithmetic calculations for the absolute angle.

$$cos\beta = \frac{R_{TB}.X}{||R_{TB}||.||X||} = \frac{\begin{bmatrix} S\theta C\phi \\ -S\phi \\ 0 \end{bmatrix} . \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}{||R_{TB}||.||X||} \tag{6.9}$$

$$cos\beta = \frac{S\theta C\phi}{\sqrt{(S\theta C\phi)^2 + (-S\phi)^2}} \tag{6.10}$$

Now, pitch and roll offset values are the resource for calculating the inclination and orientation angles of the soft neck platform.

6.3 Experimental results with the spring-based neck

This section presents the experimental results obtained from the spring-based neck and discusses the performance of the platform and the accuracy of its mathematical model.

The first test carried out consists of moving the neck two rounds of 360° with steps of 10° in orientation and 15° for the first round and 25° for the second round in inclination. The time between steps is 1s. Inner control loops are closed for the motors using the fractional order controller in Equation (6.4).

In addition, load tests are performed with $0gr$, $500gr$, and $1000gr$ masses placed on the top of the platform. The resulting performance of the neck is shown in the video available at https://vimeo.com/818675230.

The working condition of the system in this test is as follows: the input reference (neck pose trajectory) is computed by the inverse kinematic model, which determines the corresponding tendon lengths. These lengths are transformed into reference angular positions for each motor, and the inner motor control loops are responsible for commanding the motors to accurately reach the required angular positions. If the kinematic model is accurate and the spring dynamics is neglegible, as assumed, then the final real pose of the neck, measured by the IMU sensor, will be ideally equal to the reference pose.

Focusing first on the motor control loop, Figure 6.5 shows the reference angular position for motor 1 obtained from the inverse kinematic model and the resulting motor position outputs measured from its encoder corresponding to the different loads used in the experiment. For the sake of simplicity, only the analysis of motor 1 signals is shown, since the rest of motors show

a very similar response.

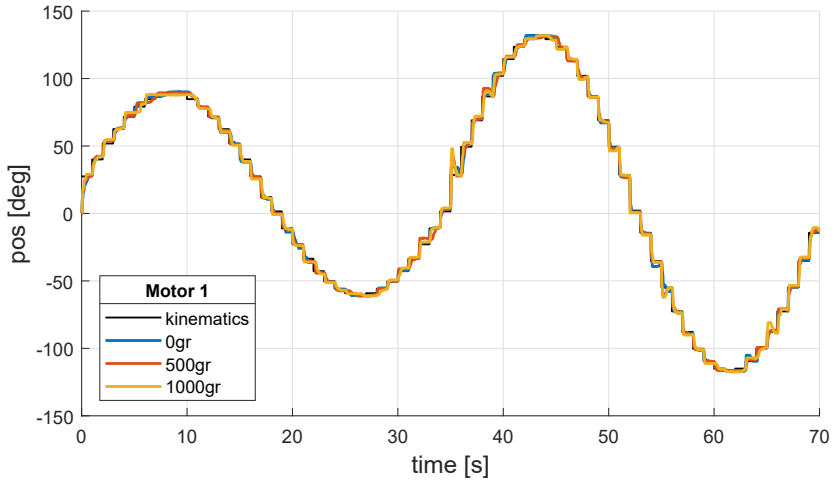


Figure 6.5: Motor 1 encoder position signals for the different payloads of $0gr$, $500gr$, and $1000gr$.

It can be seen that the inner motor control loop performs robust and accurately, with negligible position errors.

Addressing now the direct kinematic problem (as described in Section 4.1.4), Figures 6.6 and 6.7 show, for the specific neck pose trajectory (black dashed lines), the inclination and orientation neck angles obtained by applying the direct kinematic model to the real motor positions measured by the encoder (blue curves). Both figures shows a strong match between target and real (kinematics) poses.

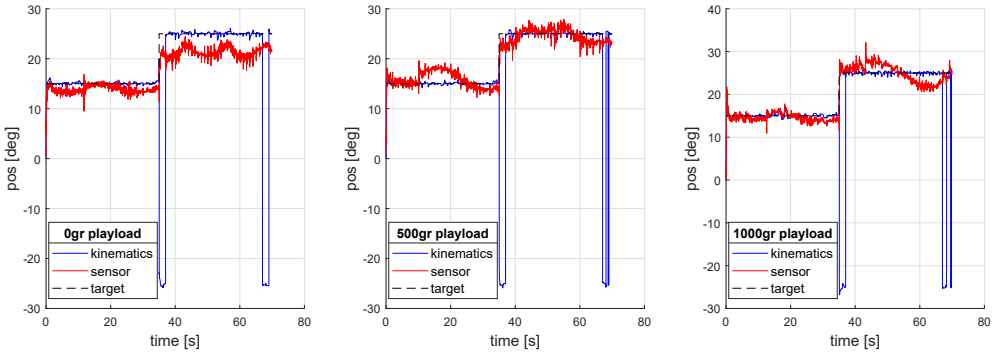


Figure 6.6: Inclination angle measured from the encoder-based kinematic model (blue curve) and from the IMU sensor (red curve) for the different payloads of $0gr$, $500gr$, and $1000gr$.

Some signal peaks can be observed in the figures. They correspond to the transient moments when a new pose trajectory round starts and the orientation of the neck jumps from 360º to 0º. The sensor resolution is not high enough to avoid this transient measurements.

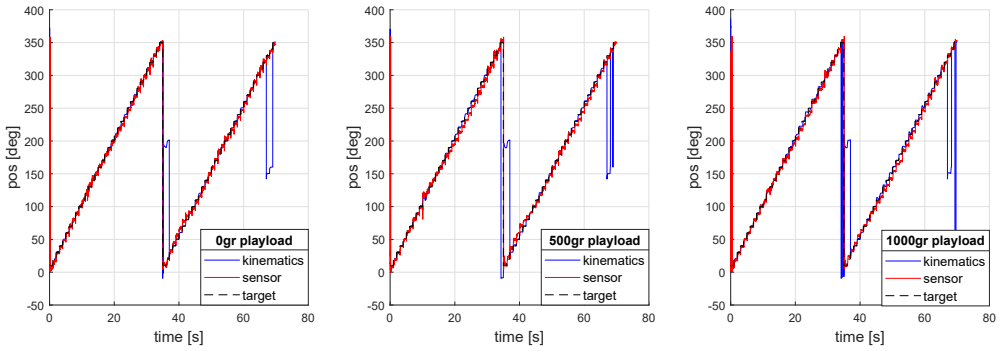


Figure 6.7: Orientation angle measured from the encoder-based kinematic model (blue curve) and from the IMU sensor (red curve) for the different payloads.

Even if the kinematic model of the spring-based neck has proven to be valid and the platform performs according to the design requirements, it is clear that the spring dynamics cannot be neglected, since it affects the final neck pose, as the red curves in Figures 6.6 and 6.7 show corresponding to the IMU sensor measurements. It can be seen that the IMU measurements indicate that an error exists between the real neck pose (red sensor curve) and that calculated from the encoder-based kinematic model (blue kinematics curve). The error is not as significant for the orientation angle as it is for the inclination angle, as shown in the figures.

Figure 6.8 shows how the neck load affects the final pose. For the same inclination target, a slightly higher inclination is caused for higher loads, which cannot be sufficiently corrected with the encoder-based inner control loop proposed.

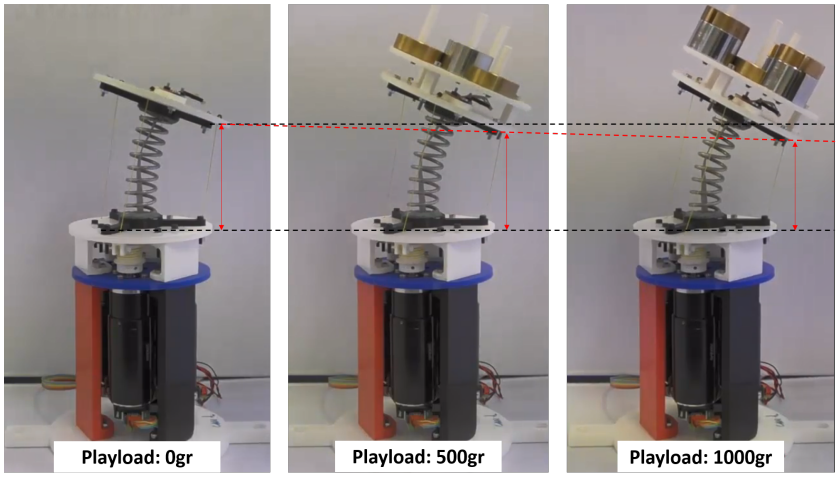


Figure 6.8: Experimental tests of the spring-based neck with different payloads.

It is clear that, in order to improve the neck performance, an outer IMU-based control loop is needed so that the dynamics of the beam is considered in the performance [97]. In the following sections, where the behaviour of the soft link-based neck is analysed, this control

strategy will be implemented.

6.4 Experimental results with the soft link-based neck

In this section the performance and kinematic model of the soft link-based neck is evaluated.

First of all, the MPU-9250 IMU sensor is installed on the top of the platform and calibrated. Then, the same encoder-based inner motor control loop used in the spring-based neck is closed to guarantee a robust motor performance. Then, the neck is commanded to reach an orientation of 90° and inclination steps of 10°, 20°, and 1°, with payloads of $0gr$, $600gr$, and $1000gr$ placed on the top of the platform. The performance is measured using the encoders and the IMU sensor, as shown in Figure 6.9 for the inclination and Figure 6.10 for the orientation. A video of the test can be seen at https://vimeo.com/821297179.

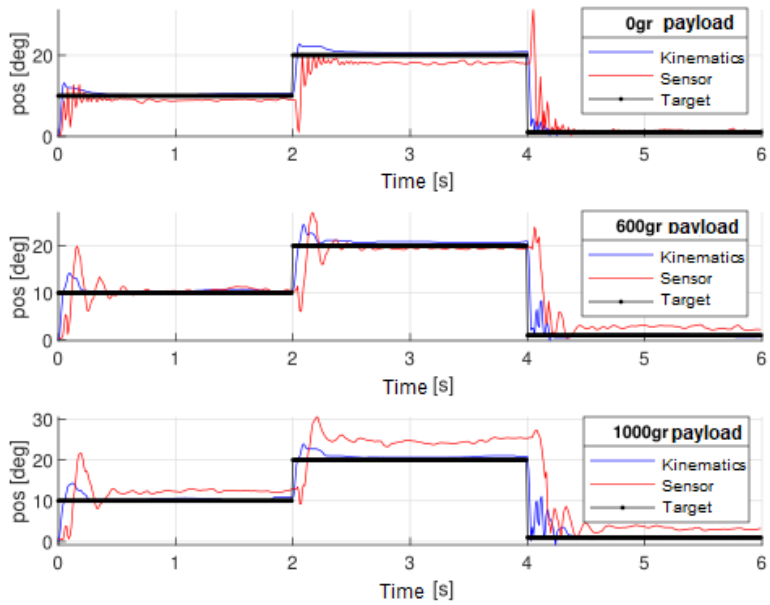


Figure 6.9: Inclination measured through the IMU sensor (red curve) and through the encoder-based kinematic model (blue curve) for different payloads of $0gr$, $600gr$, and $1000gr$.

Figure 6.9 shows that the IMU measurements differ, although not very significantly, from those estimated by the motor encoders using the kinematic model. The differences are smaller for the $0gr$ and $600gr$ loads, and a bit higher for the $1000gr$ load, where the actual inclination measured by the IMU is up to 5.3° above the estimated inclination measured by the encoders.

In the case of the orientation, the IMU measurements are very similar to those estimated

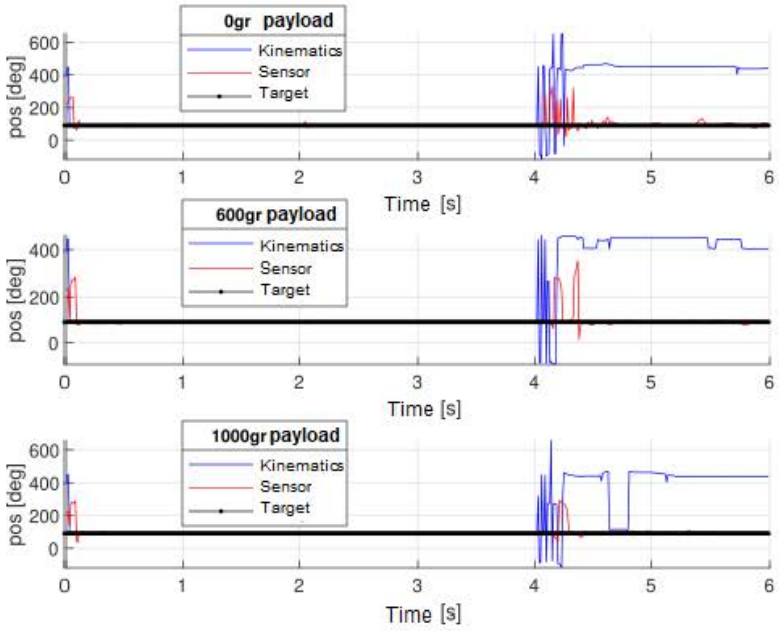


Figure 6.10: Orientation measured through the IMU sensor (red curve) and through the encoder-based kinematic model (blue curve) for different payloads of $0gr$, $600gr$, and $1000gr$.

by the encoders, with a negligible error around 90° of orientation. However, the calculation of the orientation angle from the kinematic model shows oscillating values for inclinations close to 0°, since for this range of movements the orientation can take arbitrary values between 0° and 360° (as shown by the IMU sensor, too).

According to these results, it can be concluded that the platform performs robustly and that the kinematic model of the soft link-based neck is rather accurate.

It is also clear that the encoder-based inner motor control loop is not enough to cancel the position error of the neck, specially for the case of big payloads. It is then necessary to close an IMU-based control loop that takes into account the dynamics of the beam, as shown in the following section.

6.4.1 IMU-based control approach

In this section an IMU-based control loop is implemented in the soft link-based platform following the control scheme shown in Figure 6.11, which is deeply addressed in [111].

The controller used is a fractional order FOPI tuned using the iso-m method described in [109], considering the iso-damping property of the response and the frequency specifications $\omega_c = 20 rad/s$ and $\phi_m = 45°$. The resulting FOPI controller has the form in Equation (6.11),

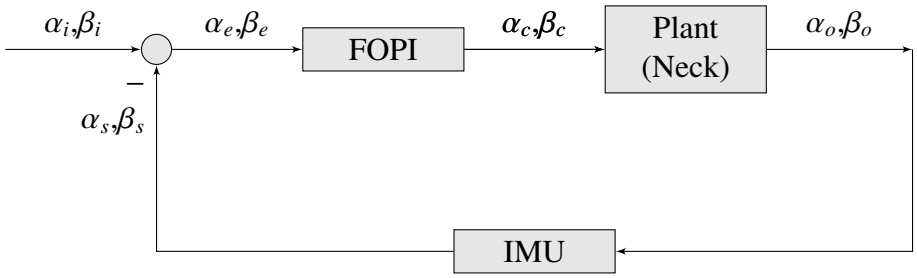


Figure 6.11: IMU-based fractional order control of the soft link-based neck.

with $k_p = 2.99$, $k_a = 22.744$ and $\alpha = -0.9$.

$$C(s) = k_p + k_a s^{\alpha} \tag{6.11}$$

The Bode diagram of the open loop system with this controller and its closed loop step response are shown in Fig. 6.12. A flat phase slope at the required gain crossover frequency can be seen in the figure, which also shows that the phase margin requirement is met.

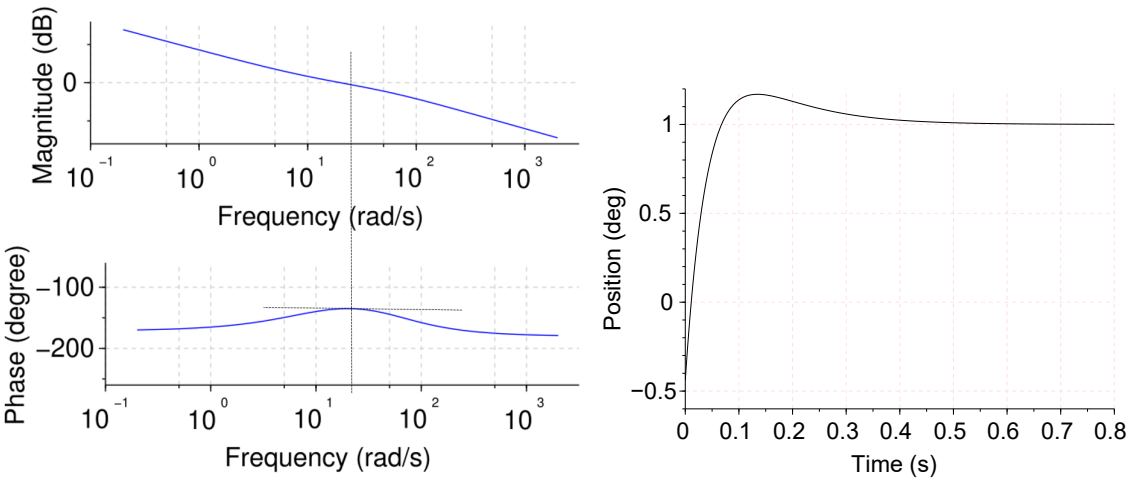


Figure 6.12: Bode diagram (left) and step response (right) of the FOPI control system.

The soft neck used for this experimentation is the refined prototype finally used for its integration into the humanoid robot TEO, which will be described in detail in the following section. The same IMU sensor is used and the inner motor control loops are simply retuned in order to adjust the performance of the new motors (now placed horizontally to reduce the neck size and facilitate its integration into the robot).

The pose trajectory used for the experiment consists of inclination steps of 15° for orientation steps of 0°, 22.5°, 45°, 67.5°, and 90°. The experiment is repeated for different payloads of $0gr$, $100gr$, $200gr$, $300gr$, $400gr$, and $500gr$.

The video showing the neck performance can be seen at https://vimeo.com/366463869,

and the results are shown in Figure 6.13 for the inclination and Figure 6.14 for the orientation.

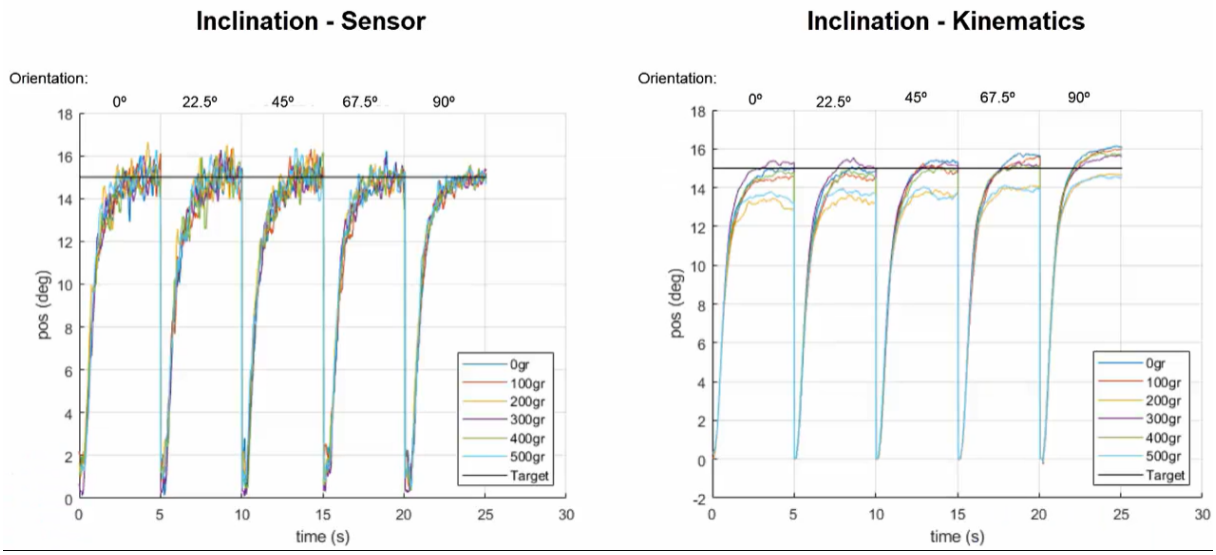


Figure 6.13: Inclination measured by the IMU sensor in closed loop (left). Inclination estimated by the encoders using the kinematic model of the neck (right). Tests for different payloads of $0gr$, $100gr$, $200gr$, $300gr$, $400gr$, and $500gr$.

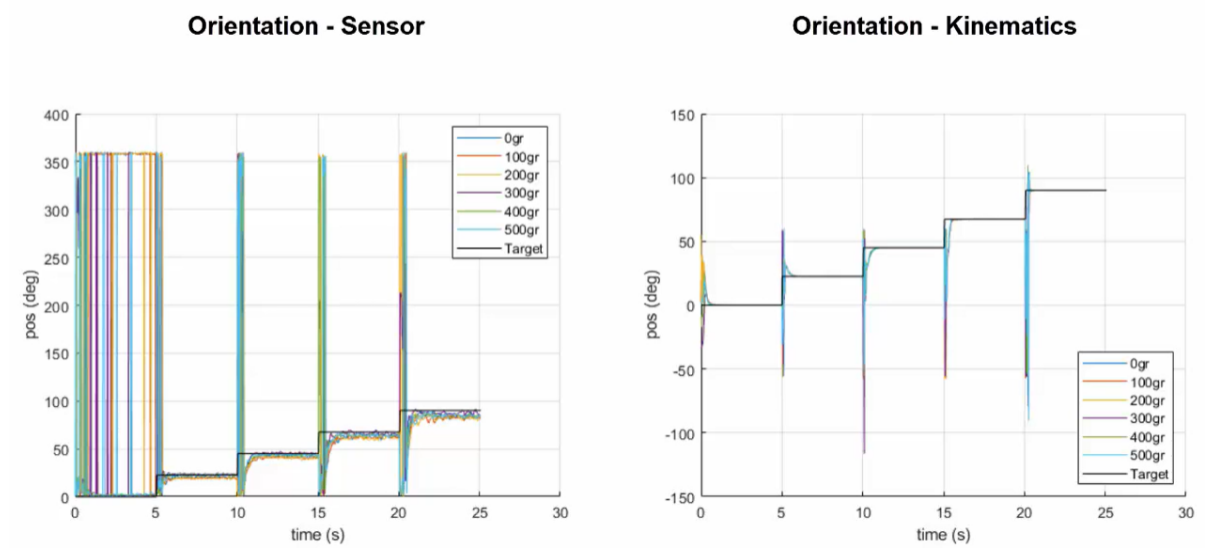


Figure 6.14: Orientation measured by the IMU sensor in closed loop (left). Orientation estimated by the encoders using the kinematic model of the neck (right). Tests for different payloads of $0gr$, $100gr$, $200gr$, $300gr$, $400gr$, and $500gr$.

Note again that orientation peaks from 0° to 360° (Figure 6.14) appears at transition poses where the inclination reaches angles very close to 0°, as discussed for the case of the spring-based neck.

The IMU results (left figures) show that the neck in closed loop performs robustly to load changes and that the inclination and orientation targets are reached with negligible errors and an

overdamped response in all cases. A possible reason for the difference between the experimental time response measured by the IMU and the simulation response could be the accuracy of the model used in simulation. It is possible that the mathematical model underestimates the stiffness of the real system, resulting in an underdamped response in the simulation. In addition, there may be factors not considered in the model, such as friction and soft material nonlinearity, which affect the response of the real system and are not reflected in the simulation.

Figures also show that the encoder-based inner motor control loops are not sufficient to cancel the position errors (right figures), specially for the inclination case.

6.5 Experimental results with the soft link-based neck integrated in the humanoid robot TEO

This section presents the performance of the soft link-based neck integrated in the humanoid robot TEO. For the sake of integration, the original neck prototype has been refined, scaling some parts and redesigning others to allow a better integration.

Figure 6.15 shows the final soft neck ready for its integration into the robot. The platform includes the connection board, the connection cable between the IMU sensor and the Arduino board, and the power and CAN cables. The platform is assembled to the robot chest through the mechanical piece shown in Figure 6.16, that allows an easy attachment of the neck. A deeper analysis of this final design has been published in [112]. Figure 6.17 shows a picture of the soft neck connected to the humanoid robot TEO.

Once the neck has been integrated into the humanoid, a lot of experiments have been performed to validate different control approaches, which are out of the scope of this thesis. Some representative videos are available here: https://vimeo.com/756294204, https://vimeo.com/435770880, and https://vimeo.com/430811271.

6.6 Experimental validation of the machine learning-based model

In this section, the MLP models discussed in Chapter 5 will be experimentally validated.

The test performed for validation consists of commanding the refined soft link-based neck

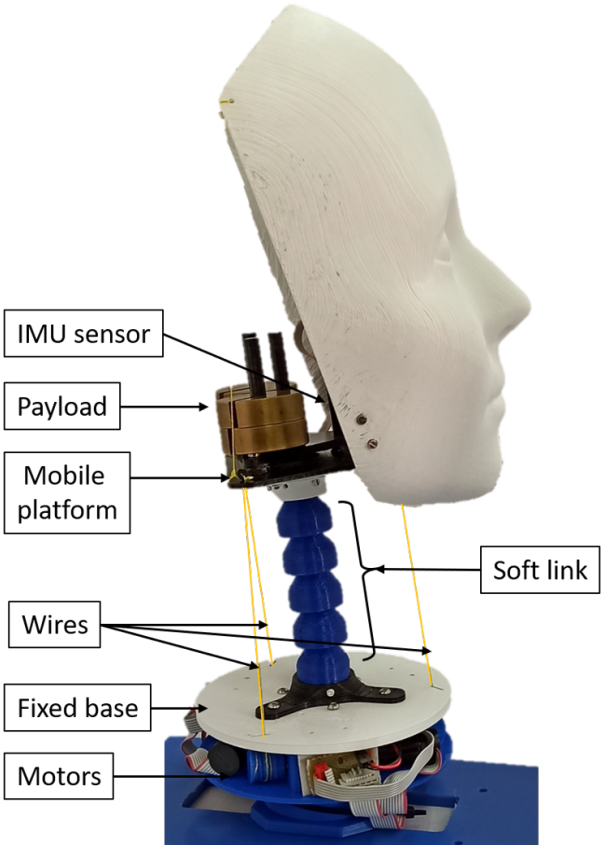


Figure 6.15: Completely assembled neck platform, including the connection board, the connection cable between the IMU sensor and the Arduino board, and the power and CAN cables.

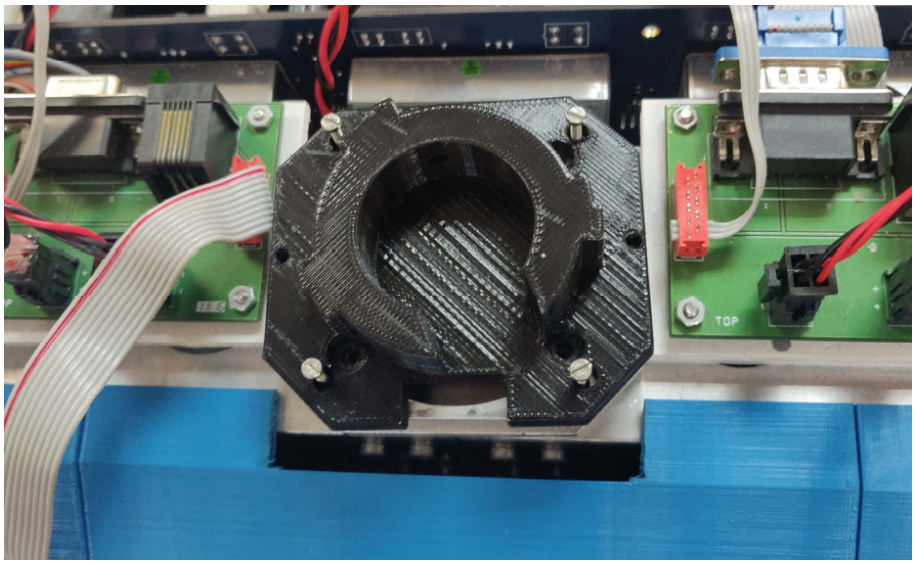


Figure 6.16: Mechanical piece to attach the soft robotic neck to the robot's chest.

to track a pose trajectory that reaches inclinations from 5° to 35° in steps of 5°, and for each of these inclinations, the orientation of the neck varies from 0° to 360° in steps of 10°, as described in Section 5.3.1.

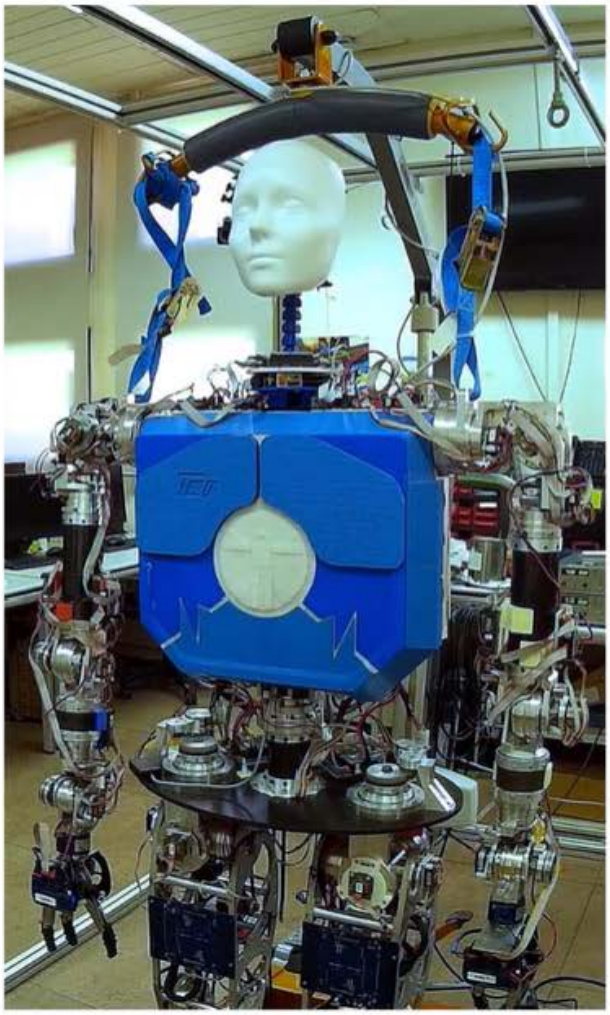


Figure 6.17: Soft neck fully integrated into the humanoid robot TEO.

This trajectory is commanded using the analytical inverse kinematics model described in Section 4.2.1. The resulting actual workspace reached by the neck, as measured by the IMU sensor, is the one shown in Figure 6.18 and will be used as reference to validate the data-driven models.

For a better analysis of the results, the inclination and orientation angles measured by the IMU sensor are presented in Figure 6.19. For the orientation, a good fit of the analytical kinematics model is observed. However, it can be seen that the actual inclination angle differs from the commanded one, and the difference increases as the value of the desired inclination increases. We will compare now this behaviour with that of the MLP models.

Now the two different workspaces obtained by the application of the two modified MLP models presented in Table 5.2 are shown respectively in Figure 6.20, for the MLP with constant architecture and TANH activation function, and Figure 6.21, for the MLP with constant architecture and ELU activation function. The pose trajectory is now commanded by using the motor

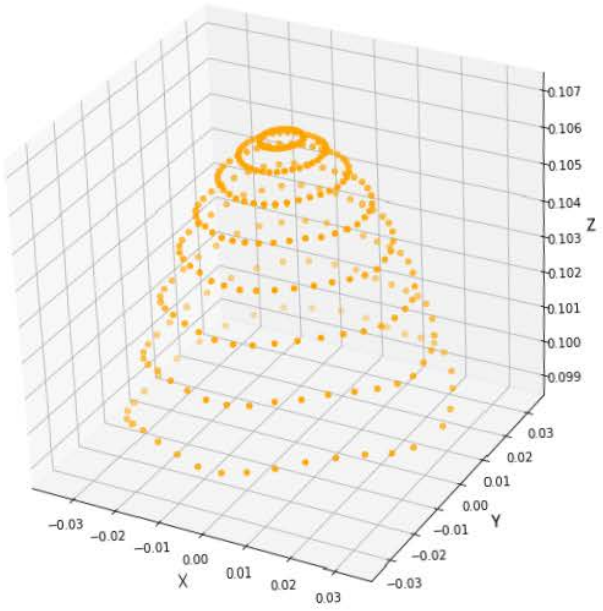


Figure 6.18: Neck workspace - Analytical inverse kinematic model.

angular positions learnt by each data-driven model, and the actual pose is measured using the IMU sensor.

From the analysis of the results shown in Figure 6.20, it is concluded that the model based on the TANH constant architecture does not correctly estimate the values of the angular positions of the motors, resulting in a workspace far from the desired one and with significant errors both in orientation and inclination, as seen in Figure 6.22. This network may have had problems predicting certain values as the TANH activation function is prone to gradient fading [113]. This occurs when the gradient is too small and tends to 0. As we move backwards during backpropagation (the way neural networks are trained), the gradient continues to get smaller, causing earlier layers of the network to learn slower than later layers, leading to overfitting.

The results presented in Figure 6.21 for the ELU model show a closer approximation to the desired workspace, proving that this model estimates the values of the angular positions of the motors more accurately. This is also reflected in the trajectories shown in Figure 6.23 corresponding to the inclination and orientation angles measured by the IMU sensor. A smaller deviation is observed in comparison with the TANH model, but higher in comparison with the analytical inverse kinematics model.

Finally, the performance of the optimized MLP model shown in Table 5.4 is discussed, based on hyper-parameters optimization. Figure 6.24 shows the workspace resulting from the application of the angular positions returned by this model. It can be seen that this network

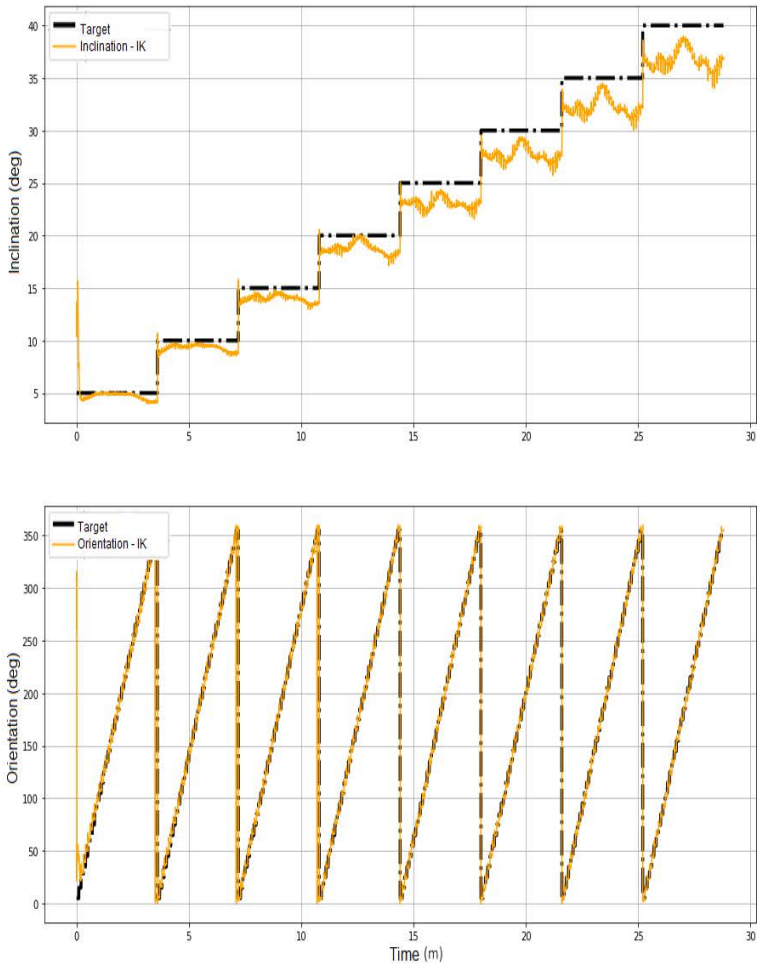


Figure 6.19: Inclination and orientation measured by the IMU - Inverse kinematics model.

is more accurate in predicting the inverse kinematic model than the constant TANH network and the constant ELU network when compared to the results shown in Figures 6.20 and 6.21, respectively. Similarly, Figure 6.25 shows a more accurate approximation than the other models in terms of the desired inclination and orientation values, although the fitting is not quite fine for some inclination values.

Figure 6.26 shows the errors (in percentage) obtained when comparing the inclinations of each of the poses obtained by the four models for the trajectory proposed in the experiment. It can be observed that the least significant error is that of the optimized MLP model, since its value tends to zero for most of the inclination values. As for the orientation errors, it has been shown throughout the discussion of the results that they are negligible, since the difference between the desired and real values is almost zero.

Two main conclusions can be remarked from this analysis: 1) the optimized MLP approach is valid for its application to the modeling of the soft neck; 2) the analytical inverse kinematics

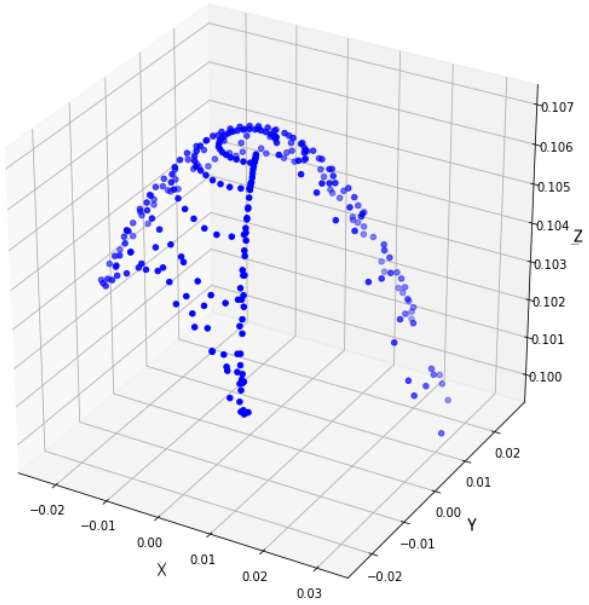


Figure 6.20: Neck workspace - TANH activation model.

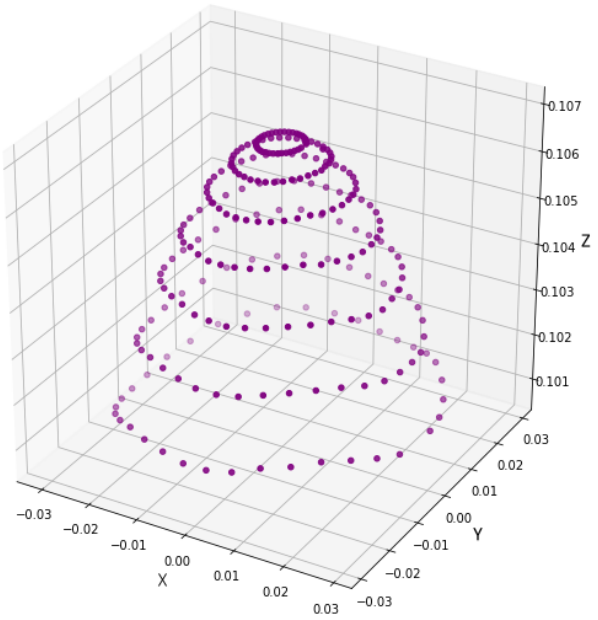


Figure 6.21: Neck workspace - ELU activation model.

model has proven to be accurate and valid for the experimental performance of the soft neck.

6.7 Conclusions

In this chapter, the designs and models of the different soft necks proposed in this thesis have been tested and validated. Various tests have been carried out to evaluate their performance and integration on the humanoid robot TEO. The experimental results demonstrate the robust-

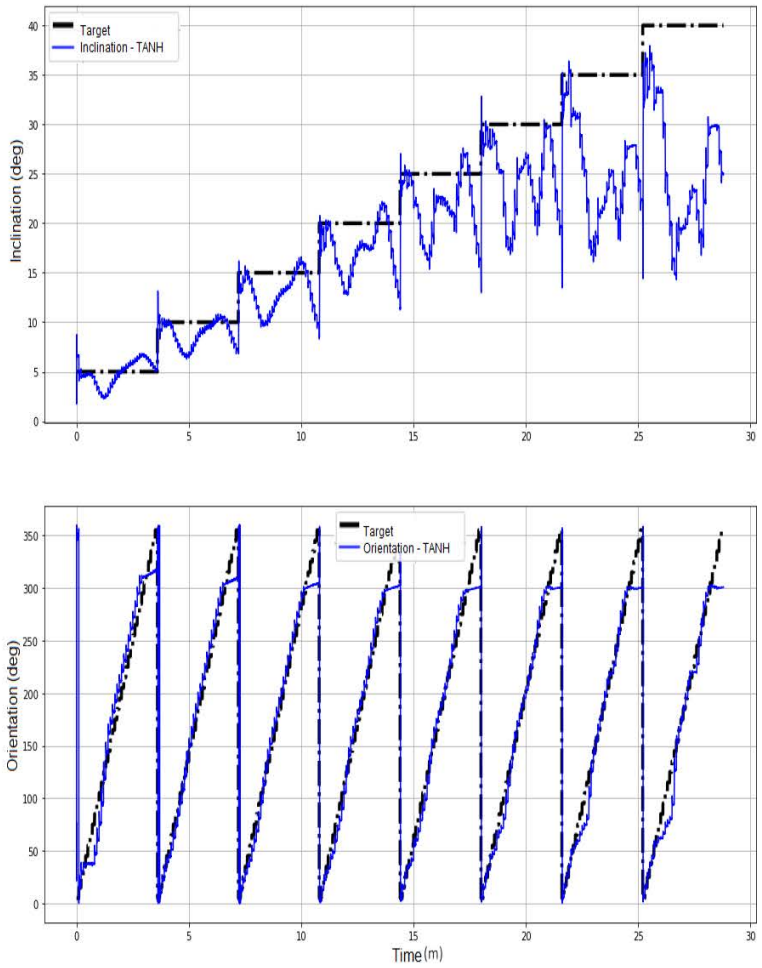


Figure 6.22: Inclination and orientation measured by the IMU - TANH activation model.

ness and accuracy of the soft neck designs, both in standalone tests and when integrated into the robot. The validation of the machine learning-based model has also shown its potential for accurate estimation of the neck angular positions. These findings contribute to the development of effective and reliable soft neck designs for humanoid robots.

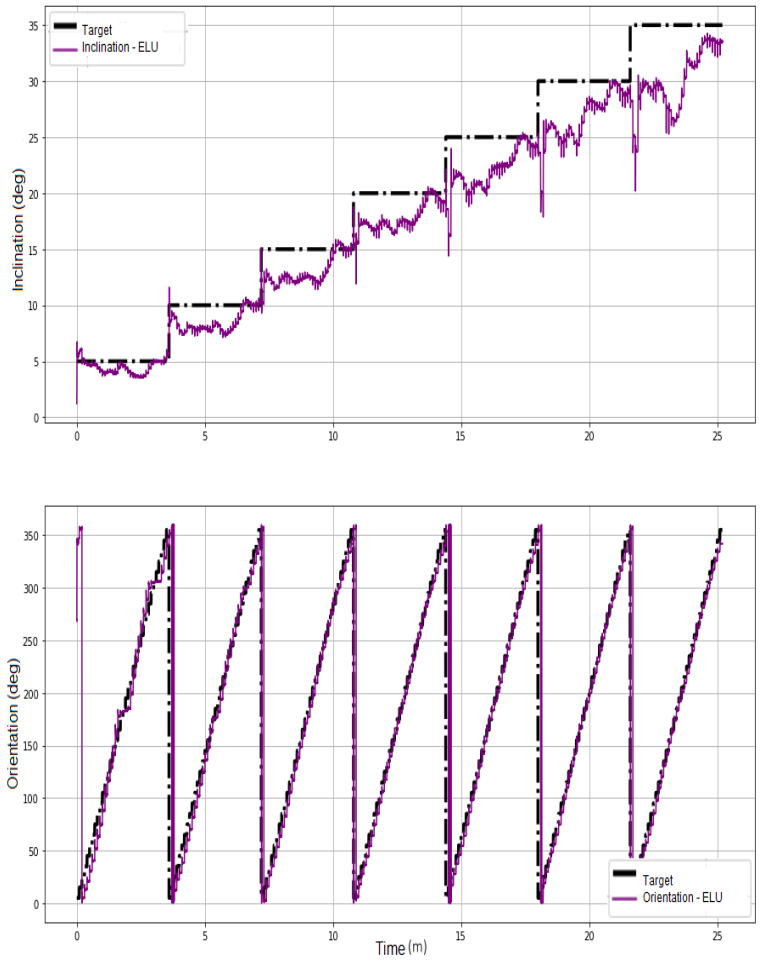


Figure 6.23: Inclination and orientation measured by the IMU - ELU activation model.

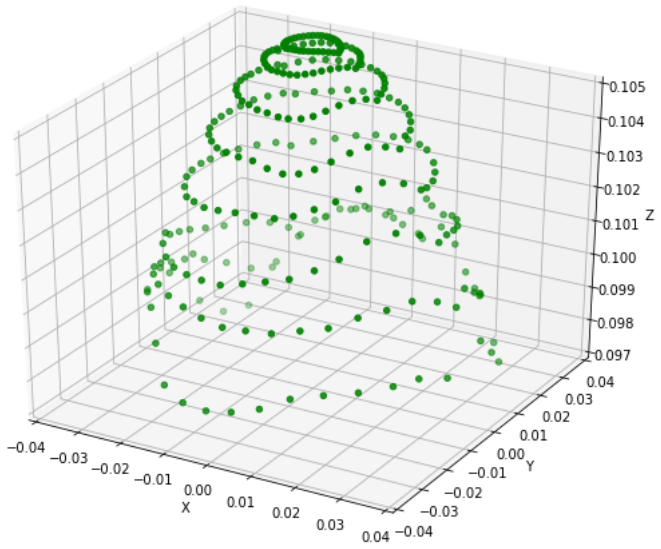


Figure 6.24: Neck workspace - Optimized MLP model.

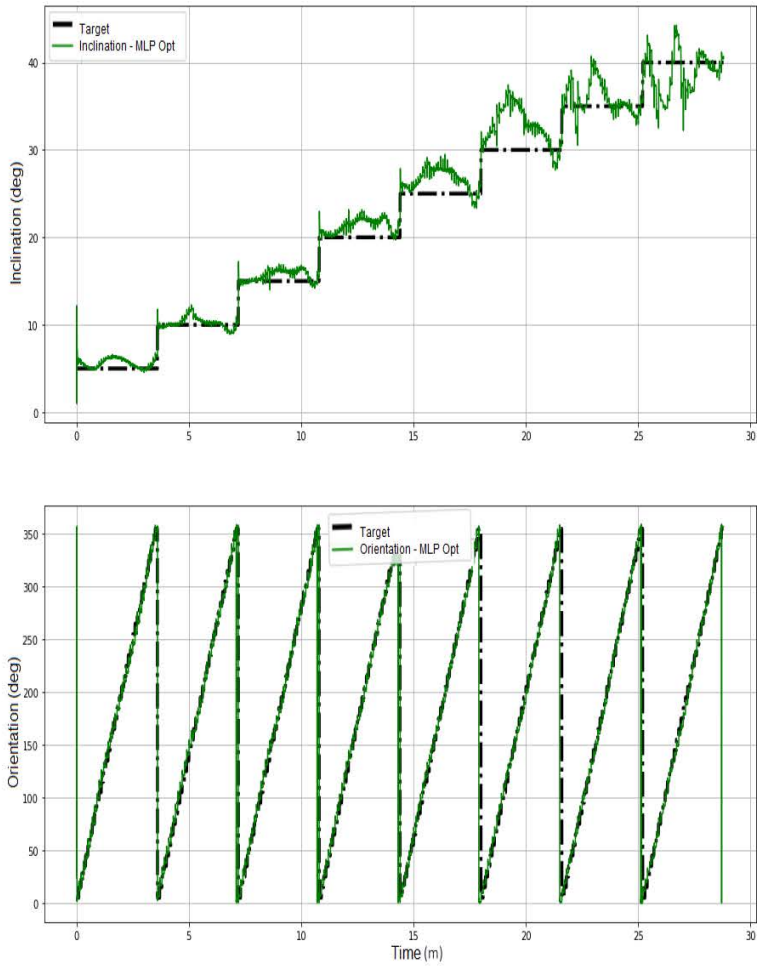


Figure 6.25: Inclination and orientation measured by the IMU - Optimized MLP model.

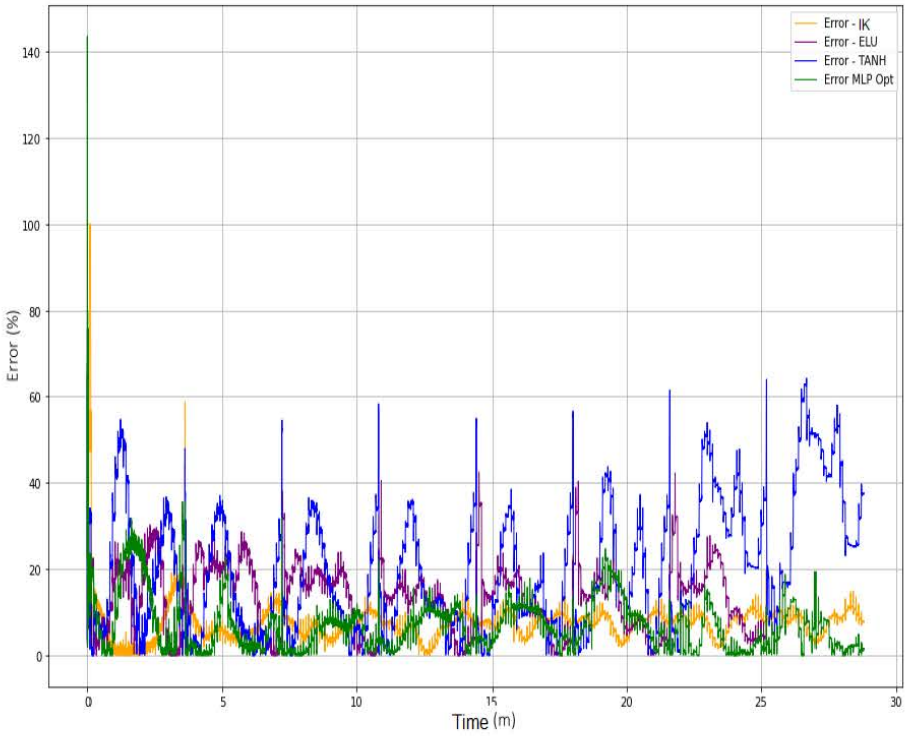


Figure 6.26: Inclination error (in percentage) for the four models tested.

Chapter 7

CONCLUSIONS

This section outlines the main conclusions derived from the research carried out in this thesis. The conclusions are organized according to the global objectives proposed in Chapter 1. The main contributions of this doctoral thesis are also presented. Finally, future research lines following this work are described.

7.1 Conclusions

Soft robotics is increasingly opening up new possibilities in many fields, including humanoid robotics. This thesis has explored the potential in collaboration between these two fields, stressing the crucial role of soft robotics to improve the adaptability, flexibility, and safety of humanoid robots. The research described in this thesis provides a significant contribution to soft humanoids through the development of a soft joint for the neck of the humanoid robot TEO. This joint illustrates the soft robotics potential to improve the interaction between a humanoid robot and its environment, allowing a smoother and more natural movement.

It is worth to mention that this thesis addresses the objectives proposed in the HumaSoft project, therefore the research described was started from scratch. Developing the initial mechanical design was a major challenge, requiring several iterations to achieve a functional and simple design, as proposed in the objectives. This was a challenging and complex task; however, these iterations were essential to achieve the desired outcome and represent an important component of this research.

The initial design of the robotic neck was based on a CDPM mechanism with two DOF and resulted in a simple and easily transportable prototype with a weight of 2kg with a payload

capacity of 1kg. Subsequently, several redesigns were carried out in order to integrate the neck into the humanoid robot. The result was a significant weight reduction to 150g, while maintaining the ability to support a payload of 1kg, leading to a huge increase in the robot's efficiency in terms of weight-to-payload ratio.

The CANOpen electronic and software designs developed for the first spring-based neck were continued in the subsequent neck and soft arm designs, with some minor modifications. It is important to note that these software and hardware systems are compatible with those used in TEO, which simplifies the integration and use of these systems in subsequent designs.

Developing a mathematical model for the soft neck was a challenge, as the designed mechanisms include a flexible nonlinear element as a backbone, making the modeling process quite difficult. Both the geometry and the material of these elements include many difficult-to-quantify factors. In the case of the spring-based neck, the mechanical properties were provided by the manufacturer, but the soft neck was CAD-designed and then manufactured in flexible material using a 3D printer. In this case, different factors (e.g. ambient temperature) influence both the process of printing and the neck operation, and therefore it is very difficult to quantify the mechanical performance parameters. Nevertheless, the final modeling accuracy is pretty good, allowing to design controllers and perform simulations, essential for robot design and control.

A mathematical model based on geometric proportions was developed for the soft neck in order to simplify the calculations. However, the nature of the soft elements of the robot makes it difficult to model its kinematics. As observed in experimental tests with an IMU sensor, despite having a mathematical model, the desired poses are not accurately achieved. Therefore, the use of machine learning-based modeling techniques such as MPL neural networks were proposed. As demonstrated in this thesis, these data-driven models were a valid and useful tool to identify the inverse kinematics of the soft neck, as seen in the experiments.

Different types of net architectures were proposed and tested. At first, a neural network with a TANH activation function was proposed, proving to be insufficient to identify the neck model, as only certain regions of its workspace can be identified according to the inverse kinematics. Based on that results, a network with an ELU activation function was used, showing a better network performance with a more accurate identification of the workspace. With the activation function defined, other parameters such as the number of hidden layers and the number of neurons in each layer have been optimized using a hyper-parameter MLP network. The

optimum parameters for the neck model were found using the GridSearchCV function of the Scikit-learn library.

After an extensive set of experiments, the evidence shows that the optimized MLP net has an excellent performance, providing high prediction accuracy throughout the whole operating neck range, providing a real inverse kinematics model of the neck.

Furthermore, it is important to note that although an acceptable accuracy value is obtained during the training of the different architectures, it is not guaranteed that such an architecture will have the same accuracy when evaluated on the soft neck in subsequent experiments. Therefore, a careful analysis of the neural network architectures used is required to ensure an accurate identification of the mathematical model of the inverse kinematics of the soft robotic neck.

Based on the knowledge achieved through the soft neck development, a soft arm was also developed with the purpose of equipping the humanoid robot TEO with more soft limbs. The challenge in the soft arm design lays in considering the effect of gravity. That was not a problem in the design of the neck, as the payload applies in the same direction as gravity forces. An innovative design of a flexible and compact soft link was proposed to solve this problem, which is currently under a licensing process. The arm morphology allows a better external loads and gravity management due to a locking configuration. The proposed design is tendon-driven and has a morphology with two main bending configurations, which provides more versatility and a bending limit compared to previous designs. In addition, these features and configurations can be modified by joint morphology parameters to achieve different working ranges and functionalities. Modeling and control issues of this soft arm are being currently addressed in another running thesis.

7.2 Future Works

Given that this thesis is mainly focused on the mechanical design and modeling of soft joints to be integrated in humanoid robots, there is still a good number of issues that need to be solved in the future.

One of these issues is the joint position control. The inverse kinematics accuracy is limited as discussed previously; therefore, the robotic necks need a solution capable to reach any desired position or orientation despite modeling errors or nonlinear behaviors. Some preliminary works like the fractional order robust control proposed in [109] have been found able to solve this

problem. Later, that approach was used to propose similar solutions to different platforms, as described in [96], [97], [111], [114], [115] with very good results. Although that strategy can effectively solve the positioning problem, there is still an important work to do to improve the controllers proposed. New techniques like Model Free Control or Model Predictive Control could improve the performance of the soft joints designed in this thesis.

Another important line of development is the possible applications emergent from the inherent soft behavior of the neck. In that direction, some works have been already proposed, like a camera stabilization application or a human gesture replication for the humanoid robot TEO. But given the potential of the soft joints proposed, a future line of research would be to look for more practical applications, which will probably involve redesigning the joint to make it adaptable to such applications.

Another important issue to be addressed is the soft neck load estimation. This approach would use the system identification based on the instantaneous input and output data, captured during the robot operation, improving the accuracy and efficiency of the neck control and its ability to perform specific functions in different environments and situations.

Similar to the soft neck, the soft arm development provides many future research possibilities, beginning with its integration into the humanoid robot TEO. This integration will probably involve an intense redesign of the electromechanical systems, but will undoubtedly improve the ability of robot to interact with the environment and perform a wide range of complex tasks.

Bibliography

[1] C. Laschi, "Octobot - a robot octopus points the way to soft robotics," *IEEE Spectrum*, vol. 54, no. 3, pp. 38–43, 2017.

[2] C. Majidi, "Soft-matter engineering for soft robotics," *Advanced Materials Technologies*, vol. 4, no. 2, p. 1 800 477, 2019.

[3] E. Sachyani Keneth, A. Kamyshny, M. Totaro, L. Beccai, and S. Magdassi, "3D printing materials for soft robotics," *Advanced Materials*, vol. 33, no. 19, p. 2 003 387, 2021.

[4] B. Trimmer, "Soft robots," *Current Biology*, vol. 23, no. 15, R639–R641, 2013.

[5] N. Elango and A. A. M. Faudzi, "A review article: Investigations on soft materials for soft robot manipulations," *The International Journal of Advanced Manufacturing Technology*, vol. 80, pp. 1027–1037, 2015.

[6] L. Marechal, P. Balland, L. Lindenroth, F. Petrou, C. Kontovounisios, and F. Bello, "Toward a common framework and database of materials for soft robotics," *Soft robotics*, vol. 8, no. 3, pp. 284–297, 2021.

[7] S. S. Robinson *et al.*, "Integrated soft sensors and elastomeric actuators for tactile machines with kinesthetic sense," *Extreme Mechanics Letters*, vol. 5, pp. 47–53, 2015.

[8] J. Li, M. Sun, and Z. Wu, "Design and fabrication of a low-cost silicone and water-based soft actuator with a high load-to-weight ratio," *Soft Robotics*, vol. 8, no. 4, pp. 448–461, 2021.

[9] Y. Hao *et al.*, "Modeling and experiments of a soft robotic gripper in amphibious environments," *International Journal of Advanced Robotic Systems*, vol. 14, no. 3, pp. 1–12, 2017.

[10] J. Jørgensen, K. B. Bojesen, and E. Jochum, "Is a soft robot more "natural"? Exploring the perception of soft robotics in human–robot interaction," *International Journal of Social Robotics*, pp. 1–19, 2022.

[11] J. Z. Gul *et al.*, "3D printing for soft robotics–a review," *Science and technology of advanced materials*, vol. 19, no. 1, pp. 243–262, 2018.

[12] S. Terryn *et al.*, "A review on self-healing polymers for soft robotics," *Materials Today*, vol. 47, pp. 187–205, 2021.

[13] T. Wallin, J. Pikul, and R. F. Shepherd, "3D printing of soft robotic systems," *Nature Reviews Materials*, vol. 3, no. 6, pp. 84–100, 2018.

[14] R. Bouaziz *et al.*, "Elastic properties of polychloroprene rubbers in tension and compression during ageing," *Polymers*, vol. 12, no. 10, p. 2354, 2020.

[15] P. Maeder-York *et al.*, "Biologically inspired soft robot for thumb rehabilitation," *Journal of Medical Devices*, vol. 8, no. 2, 2014.

[16] Y. Wang, C. Gregory, and M. A. Minor, "Improving mechanical properties of molded silicone rubber for soft robotics through fabric compositing," *Soft robotics*, vol. 5, no. 3, pp. 272–290, 2018.

[17] P. Polygerinos *et al.*, "Soft robotics: Review of fluid-driven intrinsically soft devices; manufacturing, sensing, control, and applications in human-robot interaction," *Advanced Engineering Materials*, vol. 19, no. 12, p. 1 700 016, 2017.

[18] Y. Xu *et al.*, "Molecular engineered conjugated polymer with high thermal conductivity," *Science advances*, vol. 4, no. 3, eaar3031, 2018.

[19] B. Ying, R. Z. Chen, R. Zuo, J. Li, and X. Liu, "An anti-freezing, ambient-stable and highly stretchable ionic skin with strong surface adhesion for wearable sensing and soft robotics," *Advanced Functional Materials*, vol. 31, no. 42, p. 2 104 665, 2021.

[20] V. Sánchez, C. J. Walsh, and R. J. Wood, "Textile technology for soft robotic and autonomous garments," *Advanced functional materials*, vol. 31, no. 6, p. 2 008 278, 2021.

[21] T. Zhang, T. Liang, X. Yue, and D. Sameoto, "Integration of thermoresponsive velcro-like adhesive for soft robotic grasping of fabrics or smooth surfaces," in *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, IEEE, 2019, pp. 120–125.

[22] N. Farrow, L. McIntire, and N. Correll, "Functionalized textiles for interactive soft robotics," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 5525–5531.

[23] D. Kim *et al.*, "Review of machine learning methods in soft robotics," *Plos one*, vol. 16, no. 2, e0246102, 2021.

[24] H. Su *et al.*, "Pneumatic soft robots: Challenges and benefits," *Actuators*, vol. 11, no. 3, p. 92, 2022.

[25] A. Gupta, C. Eppner, S. Levine, and P. Abbeel, "Learning dexterous manipulation for a soft robotic hand from human demonstrations," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2016, pp. 3786–3793.

[26] T. Wang, L. Ge, and G. Gu, "Programmable design of soft pneu-net actuators with oblique chambers can generate coupled bending and twisting motions," *Sensors and Actuators A: Physical*, vol. 271, pp. 131–138, 2018.

[27] M. Cianchetti, C. Laschi, A. Menciassi, and P. Dario, "Biomedical applications of soft robotics," *Nature Reviews Materials*, vol. 3, no. 6, pp. 143–153, 2018.

[28] T. Helps and J. Rossiter, "Proprioceptive flexible fluidic actuators using conductive working fluids," *Soft robotics*, vol. 5, no. 2, pp. 175–189, 2018.

[29] H. Jiang *et al.*, "Design, control, and applications of a soft robotic arm," *arXiv preprint arXiv:2007.04047*, 2020.

[30] B. Tondu and P. Lopez, "Modeling and control of Mckibben artificial muscle robot actuators," *IEEE control systems Magazine*, vol. 20, no. 2, pp. 15–38, 2000.

[31] S. M. Mirvakili and I. W. Hunter, "Artificial muscles: Mechanisms, applications, and challenges," *Advanced Materials*, vol. 30, no. 6, p. 1 704 407, 2018. DOI: https://doi.org/10.1002/adma.201704407.

[32] B. Kalita, A. Leonessa, and S. K. Dwivedy, "A review on the development of pneumatic artificial muscle actuators: Force model and application," *Actuators*, vol. 11, no. 10, p. 288, 2022.

[33] S. Wakimoto, J. Misumi, and K. Suzumori, "New concept and fundamental experiments of a smart pneumatic artificial muscle with a conductive fiber," *Sensors and Actuators A: Physical*, vol. 250, pp. 48–54, 2016.

[34] K. Ashwin and A. Ghosal, "A survey on static modeling of miniaturized pneumatic artificial muscles with new model and experimental results," *Applied Mechanics Reviews*, vol. 70, no. 4, 2018.

[35] B. Kalita and S. Dwivedy, "Dynamic analysis of pneumatic artificial muscle (PAM) actuator for rehabilitation with principal parametric resonance condition," *Nonlinear Dynamics*, vol. 97, pp. 2271–2289, 2019.

[36] Z. Jiao, C. Ji, J. Zou, H. Yang, and M. Pan, "Vacuum-powered soft pneumatic twisting actuators to empower new capabilities for soft robots," *Advanced Materials Technologies*, vol. 4, no. 1, p. 1 800 429, 2019.

[37] C. Tawk, G. M. Spinks, M. in het Panhuis, and G. Alici, "3D printable linear soft vacuum actuators: Their modeling, performance quantification and application in soft robotic systems," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 5, pp. 2118–2129, 2019.

[38] C. Zhang *et al.*, "Fluid-driven artificial muscles: Bio-design, manufacturing, sensing, control, and applications," *Bio-Design and Manufacturing*, vol. 4, pp. 123–145, 2021.

[39] S. Zaidi, M. Maselli, C. Laschi, and M. Cianchetti, "Actuation technologies for soft robot grippers and manipulators: A review," *Current Robotics Reports*, vol. 2, no. 3, pp. 355–369, 2021.

[40] S. D. Katugampala, K. M. Arachchi, S. Asanka, R. B. Arumathanthri, A. L. Kulasekera, and N. D. Jayaweera, "Design and characterization of a novel vacuum bending actuator and a bimorph: For preliminary use in a continuum robot arm," in *2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, IEEE, 2019, pp. 263–268.

[41] V. Vikas, E. Cohen, R. Grassi, C. Sözer, and B. Trimmer, "Design and locomotion control of a soft robot using friction manipulation and motor–tendon actuation," *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 949–959, 2016.

[42] B. B. Kang *et al.*, "Learning-based fingertip force estimation for soft wearable hand robot with tendon-sheath mechanism," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 946–953, 2020.

[43] L. Martua, M. S. Kurbanhusen, and C. N. Aung, "Investigation to enhance the actuation force transmission of cable-driven continuum manipulators with tethering lines," 2022.

[44] Z. Xiang, L. Hongwei, D. Bingxiao, C. Lu, C. Xiaoqian, and H. Yiyong, "Design and experimental validation of a cable-driven continuum manipulator and soft gripper," in *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, IEEE, 2019, pp. 1965–1968.

[45] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi, "Dynamic model of a multibending soft robot arm driven by cables," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1109–1122, 2014.

[46] Q. Wu *et al.*, "A novel underwater bipedal walking soft robot bio-inspired by the coconut octopus," *Bioinspiration & Biomimetics*, vol. 16, no. 4, p. 046 007, 2021.

[47] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.

[48] R. Pelrine, R. Kornbluh, Q. Pei, and J. Joseph, "High-speed electrically actuated elastomers with strain greater than 100%," *Science*, vol. 287, no. 5454, pp. 836–839, 2000.

[49] F. Carpi, A. Migliore, G. Serra, and D. De Rossi, "Helical dielectric elastomer actuators," *Smart Materials and Structures*, vol. 14, no. 6, p. 1210, 2005.

[50] C. Laschi, M. Cianchetti, B. Mazzolai, L. Margheri, M. Follador, and P. Dario, "Soft robot arm inspired by the octopus," *Advanced robotics*, vol. 26, no. 7, pp. 709–727, 2012.

[51] T. Umedachi, V. Vikas, and B. Trimmer, "Softworms: The design and control of non-pneumatic, 3D-printed, deformable robots," *Bioinspiration & biomimetics*, vol. 11, no. 2, p. 025 001, 2016.

[52] M. Liu, L. Hao, W. Zhang, and Z. Zhao, "A novel design of shape-memory alloy-based soft robotic gripper with variable stiffness," *International journal of advanced robotic systems*, vol. 17, no. 1, p. 1 729 881 420 907 813, 2020.

[53] R. Niiyama, "Soft actuation and compliant mechanisms in humanoid robots," *Current Robotics Reports*, vol. 3, no. 3, pp. 111–117, 2022.

[54] J. Zhou, X. Chen, U. Chang, J. Pan, W. Wang, and Z. Wang, "Intuitive control of humanoid soft-robotic hand BCL-13," in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, IEEE, 2018, pp. 314–319.

[55] L. Li, H. Godaba, H. Ren, and J. Zhu, "Bioinspired soft actuators for eyeball motions in humanoid robots," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 1, pp. 100–108, 2018.

[56] I. Mizuuchi *et al.*, "The design and control of the flexible spine of a fully tendon-driven humanoid "kenta"," in *IEEE/RSJ international conference on intelligent robots and systems*, IEEE, vol. 3, 2002, pp. 2527–2532.

[57] C. Borst *et al.*, "Rollin'justin-mobile platform with variable base," in *2009 IEEE International Conference on Robotics and Automation*, IEEE, 2009, pp. 1597–1598.

[58] J. Reinecke, B. Deutschmann, and D. Fehrenbach, "A structurally flexible humanoid spine based on a tendon-driven elastic continuum," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 4714–4721.

[59] B. Deutschmann, A. Dietrich, and C. Ott, "Position control of an underactuated continuum mechanism using a reduced nonlinear model," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, IEEE, 2017, pp. 5223–5230.

[60] B. Deutschmann, C. Ott, C. A. Monje, and C. Balaguer, "Robust motion control of a soft robotic system using fractional order control," in *Advances in Service and Industrial Robotics: Proceedings of the 26th International Conference on Robotics in Alpe-Adria-Danube Region, RAAD 2017*, Springer, 2017, pp. 147–155.

[61] J. Gunsing *et al.*, "Adaptive robotic systems design in university of applied sciences," in *MATEC Web of Conferences*, EDP Sciences, vol. 52, 2016, p. 02 007.

[62] S. Martínez de la Casa, C. A. Monje Micharet, and A. Jardón Huete, "Humanoids. Los humanos y los robots cara a cara," *i-3, La revista de la UC3M*, vol. 51, pp. 27–31, 2015.

[63] C. A. Monje and C. Balaguer, "Humasoft: Diseño y control de eslabones blandos para robots humanoides," in *Jornadas Nacionales de Robótica*, CEA-IFAC, 2017, pp. 1–6.

[64] S. Behzadipour and A. Khajepour, "Cable-based robot manipulators with translational degrees of freedom," in *Industrial Robotics: Theory, Modelling and Control*, InTech, 2006, pp. 221–236.

[65] S. Alfayad *et al.*, "Hydroid humanoid robot head with perception and emotion capabilities: Modeling, design, and experimental results," *Frontiers in Robotics and AI*, vol. 3, p. 15, 2016.

[66] C. Gosselin, *Kinematic analysis, optimization and programming of parallel robotic manipulators*. McGill University Montreal, Canada, 1988.

[67] B. Gao, N. Xi, Y. Shen, J. Zhao, and R. Yang, "Development of a low motion-noise humanoid neck: Statics analysis and experimental validation," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, IEEE, 2010, pp. 1203–1208.

[68] S. Timoshenko, *Theory of elastic stability*, 1st ed. New York McGraw-Hill Book Company, inc, 1936, Bibliographical foot-notes.

[69] A. C. Sterling, D. G. Cobian, P. A. Anderson, and B. C. Heiderscheit, "Annual frequency and magnitude of neck motion in healthy individuals," *Spine*, vol. 33, no. 17, pp. 1882–1888, 2008.

[70] I. The MathWorks, *Matlab optimization toolbox*, The year of your version, you can find it out using ver, Natick, Massachusetts, United State. [Online]. Available: https://www.mathworks.com/help/optim/.

[71] J. Braun, *Libro de fórmulas*. Editorial Maxon Academy,Sachseln, 2013, vol. 1.

[72] M. Motor, *Maxon selection program.* https://www.maxongroup.es/maxon/view/msp, [Accessed 24-Mar-2020], 2020.

[73] NinjaFlex.com, *NinjaFlex 3D Printing Filament- Technical Specifications*, https://ninjatek.com/wp-content/uploads/NinjaFlex-TDS.pdf, [Accessed 08-Mar-2023], 2020.

[74] Tecnosoft, *iPOS2401 MX-CAN product data sheet, Rep. no. iPOS2401 MXCAN, Datasheet*, https://technosoftmotion.com/wp-content/uploads/2019/10/P091.024.iPOS2401.MX_.CAN_.CAT_.UM_.0522.pdf, [Accessed 04-Mar-2020], 2018.

[75] CANopen, *Canopen - the standardized embedded network.* https://www.can-cia.org/can-knowledge/canopen/canopen/, [Accessed 1-April-2023], 2002.

[76] CiA, *Cia 402 series: Canopen device profile for drives and motion control.* https://www.can-cia.org/can-knowledge/canopen/cia402/, [Accessed 1-April-2023], 2002.

[77] N. Puente Carreño, "Estudio y puesta en marcha de prototipo de cuello blando de robot humanoide," B.S. thesis, 2018.

[78] D. Paret, *Multiplexed networks for embedded systems: CAN, LIN, flexray, safe-by-wire*. John Wiley & Sons, 2007.

[79] CiA, *Cia 301 canopen application layer and communication profile.* https://forum.opencyphal.org/uploads/short-url/mNWuvY23DYckSFoSbfam0ji3YWn.pdf, [Accessed 1-April-2023], 2002.

[80] H. Github, *Humasoft, library under cia 402 standard for device control.* https://github.com/HUMASoft/CiA402Device, [Accessed 1-April-2023], 2017.

[81] J. Carretero, R. Podhorodeski, M. Nahon, and C. M. Gosselin, "Kinematic analysis and optimization of a new three degree-of-freedom spatial parallel manipulator," *Journal of mechanical design*, vol. 122, no. 1, pp. 17–24, 2000.

[82] F.-C. Yang and E. Haug, "Numerical analysis of the kinematic working capability of mechanisms," *Journal of Mechanical Design*, vol. 116, no. 1, pp. 111–118, 1994.

[83] I. A. Bonev and J. Ryu, "Orientation workspace analysis of 6-DOF parallel manipulators," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers, vol. 19715, 1999, pp. 281–288.

[84] B. Gao, J. Xu, J. Zhao, and N. Xi, "Combined inverse kinematic and static analysis and optimal design of a cable-driven mechanism with a spring spine," *Advanced Robotics*, vol. 26, no. 8-9, pp. 923–946, 2012.

[85] R. Frisch-Fay, "Butterworths; Washington, DC: 1962," *Flexible bars*, pp. 33–44,

[86] R. J. Webster III and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

[87] B. Siciliano, O. Khatib, and T. Kröger, *Springer handbook of robotics*. Springer, 2008, vol. 200.

[88] R. Köker, T. Çakar, and Y. Sari, "A neural-network committee machine approach to the inverse kinematics problem solution of robotic manipulators," *Engineering with Computers*, vol. 30, no. 4, pp. 641–649, 2014.

[89] Y. Becerra, M. Arbulu, S. Soto, and F. Martinez, "A comparison among the Denavit-Hartenberg, the screw theory, and the iterative methods to solve inverse kinematics for assistant robot arm," in *International Conference on Swarm Intelligence*, Springer, 2019, pp. 447–457.

[90] L. Tran, Z. Zhang, S. Yeo, Y. Sun, and G. Yang, "Control of a cable-driven 2-DOF joint module with a flexible backbone," in *2011 IEEE Conference on Sustainable Utilization and Development in Engineering and Technology (Student)*, IEEE, 2011, pp. 150–155.

[91] L. Jin, S. Li, J. Yu, and J. He, "Robot manipulator control using neural networks: A survey," *Neurocomputing*, vol. 285, pp. 23–34, 2018.

[92] K. Jiokou Kouabon, A. Melingui, O. Lakhal, M. Kom, and R. Merzouki, "A learning framework to inverse kinematics of redundant manipulators," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9912–9917, 2020.

[93] T. G. Thuruthel, E. Falotico, F. Renda, and C. Laschi, "Learning dynamic models for open loop predictive control of soft robotic manipulators," *Bioinspiration & biomimetics*, vol. 12, no. 6, p. 066 003, 2017.

[94] S. B. Šegota, N. Anđelić, V. Mrzljak, I. Lorencin, I. Kuric, and Z. Car, "Utilization of multilayer perceptron for determining the inverse kinematics of an industrial robotic manipulator," *International Journal of Advanced Robotic Systems*, vol. 18, no. 4, p. 1 729 881 420 925 283, 2021.

[95] X. Wang, X. Liu, L. Chen, and H. Hu, "Deep-learning damped least squares method for inverse kinematics of redundant robots," *Measurement*, vol. 171, p. 108 821, 2021.

[96] C. Relaño, J. Muñoz, C. A. Monje, S. Martıénez, and D. González, "Modeling and control of a soft robotic arm based on a fractional order control approach," *Fractal and Fractional*, vol. 7, no. 1, p. 8, 2022.

[97] L. Mena, C. A. Monje, L. Nagua, J. Muñoz, and C. Balaguer, "Test bench for evaluation of a soft robotic link," *Frontiers in Robotics and AI*, vol. 7, p. 27, 2020.

[98] F. Nori, L. Jamone, G. Sandini, and G. Metta, "Accurate control of a human-like tendon-driven neck," in *2007 7th IEEE-RAS International Conference on Humanoid Robots*, IEEE, 2007, pp. 371–378.

[99] J. Zou, Y. Han, and S.-S. So, "Overview of artificial neural networks," *Artificial neural networks: methods and applications*, pp. 14–22, 2009.

[100] C. Perlich, "Learning curves in machine learning," in *Encyclopedia of Machine Learning*, 2010, p. 5.

[101] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *towards data science*, vol. 6, no. 12, pp. 310–316, 2017.

[102] J. Brownlee and M. L. Mastery, *Deep Learning with Python: Develop Deep Learning Models on Theano and TensorFlow Using Keras*. Machine Learning Mastery, 2017. [Online]. Available: https://books.google.es/books?id=eJw2nQAACAAJ.

[103] M. J. Zaki and W. Meira Jr, *Data Mining and Machine Learning: Fundamental Concepts and Algorithms*, 2nd ed. Cambridge University Press, 2020. DOI: 10.1017/9781108564175.

[104] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, *Fast and accurate deep network learning by exponential linear units (elus)*, 2016. arXiv: 1511.07289 [cs.LG].

[105] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[106] A. Gholamy, V. Kreinovich, and O. Kosheleva, "Why 70/30 or 80/20 relation between training and testing sets: A pedagogical explanation," *International Journal of Intelligent Technologies and Applied Statistics*, vol. 11, no. 2, pp. 105–111, 2018.

[107] I. Goodfellow *et al.*, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[108] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu-Batlle, *Fractional-order systems and controls: fundamentals and applications*. Springer Science & Business Media, 2010.

[109] J. Muñoz Yáñez-Barnuevo, "Robust control strategies based on fractional calculus for robotic platforms," Ph.D. dissertation, Departamento de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, España, 2020.

[110] R. Mahony, T. Hamel, and J.-M. Pflimlin, "Complementary filter design on the special orthogonal group so (3)," in *Proceedings of the 44th IEEE Conference on Decision and Control*, IEEE, 2005, pp. 1477–1484.

[111] L. Nagua, J. Muñoz, L. Mena, C. A. Monje, and C. Balaguer, "Robust control strategy for improving the performance of a soft robotic link," in *XLII Jornadas de Automática*, Universidade da Coruña, Servizo de Publicacións, 2021, pp. 499–506.

[112] N. Puente Carreño, "Integración de un cuello robótico blando en el robot humanoide TEO," M.Sc. Thesis, Departamento de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, España, 2020.

[113] I. J. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[114] L. Nagua, J. Muñoz, C. A. Monje, and C. Balaguer, "A first approach to a proposal of a soft robotic link acting as a neck," in *XXXIX Jornadas de Automática*, Área de Ingeniería de Sistemas y Automática, Universidad de Extremadura, 2018, pp. 522–529.

[115] J. Muñoz, C. A. Monje, L. F. Nagua, and C. Balaguer, "A graphical tuning method for fractional order controllers based on iso-slope phase curves," *ISA transactions*, vol. 105, pp. 296–307, 2020.