UCLouvain

École polytechnique de Louvain

Modelling and control of a post-stroke robot activated by pneumatic artificial muscles for upper-limb rehabilitation

Author: **Eline Koob**
Supervisors: **Benoît Herman, Renaud Ronsse**
Readers: **Stéphanie Dehem, Bertrand Tondu**
Academic year 2020–2021
Master [120] in Electro-mechanical Engineering

Abstract

Context Nowadays, many people suffer from strokes. This disease, which mainly affects older people, causes a reduction in motor skills. It is important to recover these functions in order to return to a life as normal as possible. Rehabilitation takes place through precise and regular movements performed by the patient. This is where robotics comes in. Integrating robots into the world of rehabilitation allows us to optimise treatments and obtain better final results.

Objectives A robotic arm activated by artificial pneumatic muscles, developed by Louvain Bionics and INSA Toulouse, has been designed for upper limb rehabilitation. The main objective of this thesis is to model and control the robot. The aim of the project is to find out whether this type of actuator, which is gentler than an electric motor, has sufficient advantages to enter the medical field.

Experiments An experimental process has been carried out throughout the project. Firstly, experiments were carried out to create a model corresponding to the Festo muscles, the artificial pneumatic muscles used for the robot. Then, several tests validated the performance of the implemented controller. Finally, a hybrid stiffness control, taking full advantage of the physical stiffness of the system, was implemented and evaluated.

Results A robot activated by pneumatic artificial muscles has major advantages, such as its flexibility which gives it a behaviour similar to human muscles. It also has some disadvantages that are not negligible for its future use. Some of the disadvantages are lower accuracy of position control and loud noise due to air supply to the muscles. In addition, the results concerning the hybrid control in stiffness, which may be a main advantage of pneumatic actuators compared to electric ones, are not significantly impressive. However, further research in this area would surely improve the performances, which would make this robot really interesting for the medical robotics market.

Acknowledgment

This work would not have been properly completed without the valuable help of some people. I would like to thank those who contributed to the completion of this thesis.

I would like to thank, of course, my two supervisors, *Prof. Renaud Ronsse* and *Prof. Benoît Herman*, for their unfailing investment throughout the semester. Their advice and reflections allowed me to deepen my knowledge and to progress on the project every week, in a professional and serene working environment. I couldn't have hoped for a better follow-up for the realisation of my thesis.

I thank *Prof. Bertrand Tondu* who, despite the 825 km distance between Toulouse and Louvain-La-Neuve, has shown a significant investment in this project. His expertise on the subject was of great importance.

I thank *Prof. Stéphanie Dehem* for having accepted to be a member of my jury, but also for her availability and her sharing of knowledge in the medical field. This project is the result of a collaboration between the robotics and medical worlds. One needs the other to continue innovating in the field of medical robotics.

I thank *Dr. Don Robert*, my American host father by heart, who agreed to proofread and correct my thesis because, I quote, "it is always a pleasure to help women advance in engineering and computers".

Finally, I would like to thank my family and friends who have been an important support throughout my studies. More particularly, I would like to thank *Aurélien* who has been a pillar for me from the beginning to the end. He never stopped encouraging me and pulling me up. I am deeply grateful to him for always believing in me, and for supporting me during these years.

Contents

Nomenclature

Acronyms

DoF	Degree Of Freedom
EE	End-Effector
FB	Feedback
FF	Feed-forward
I/P	Intensity/Pressure
NI	National Instrument
PAM	Pneumatic Artificial Muscle
PI	Proportional Integral
tPA	Tissue Plasminogen Activator

Control Variables

(F_x, F_y)	Handle force components
(K_x, K_y)	Handle stiffness components
τ_j	Torque developed by joint $j = 1, 2$
F_i	Force developed by the muscle $i = 1, 2, 3, 4$
K	Stiffness factor
K_j	Stiffness developed by the articulation $j = 1, 2$
K_{hard}	Hardware/physical stiffness
K_{soft}	Software stiffness
P_i	Pressure inserted into the muscle $i = 1, 2, 3, 4$

Dimensional Variables

δl	Initial shortening length of the muscle
ε_0	Initial contraction ratio of the muscle
ε_i	Contraction ratio of the muscle $i = 1, 2, 3, 4$
l	Length of the muscle
l_0	Initial length of the muscle
L_j	Length of the segment $j = 1, 2$
R	Radius of the pulley
r_0	Initial radius of the muscle

Angular Variables

α_j	Angle of the joint $j = 1, 2$ in its mobile frame
θ_j	Angle of the joint $j = 1, 2$ in fixed reference frame

Other Variables

c	Damping factor
f_c	Cut-off frequency
J	Jacobian operator
K_i	Integral gain of the PI controller
K_p	Proportional gain of the PI controller
m	Mass
Z	Impedance

Introduction

Stroke is a disease that affects 75,000 people each year, and is one of the main causes of long-term adult disability [22]. Older people are the biggest targets of this disease. Stroke rates increase exponentially with age. Between one-third and one-half of patients die of stroke and only 5% to 20% of stroke survivors fully recover after being affected [33] [22].

Stroke causes long-term physical problems, as well as cognitive consequences [4]. It also produces sensory impairments and visual and perceptual disorders. Patients' mobility, as well as their ability to read or speak may be reduced. All these impairments lead to a loss of motivation which has an negative impact on potential rehabilitation and causes anxiety and depression. It reduces quality of life and results in social isolation.

Progressive and constant rehabilitation can significantly reduce or even eliminate the cognitive and motor sequelae of stroke. Stroke and stroke recovery are largely heterogeneous [4]. They depend on many individual factors specific to each patient. Therapy should start as soon as possible after the stroke. It should lead to intensive motor practice and should include repetitive tasks in order to achieve great improvements. Active retraining is also used to shape the cortical reorganisation.

Integrating a robot into the therapy improves the quality of the recovery treatment after a stroke. It increases the effectiveness of the treatment, and lightens the physical work of the doctor. They allow a more independent training of the therapist, which makes it more ergonomic and saves time. The training provided by the robot is active and can consist of intense and repetitive tasks. The robot can provide different resistance precisely according to the person, the exercise, and even over time. They are precise and optimise the movements performed, since trajectories can be personalised for each patient. Robot-assisted therapy has advantages in terms of clinical and biomechanical measures. These devices are used both as a precise measuring instrument for position or force, and to quantify motor impairments such as spasticity, tone, and strength [14]. They allow the patient's progress to be tracked. Visual feedback can improve live performance and can also encourage the patient to work in the form of games.

In order to further contribute to the development of robotics in the medical world, a project has been developed in collaboration between Louvain Bionics, UCLouvain and INSA Toulouse. It is a robot for the rehabilitation of the upper limb for stroke victims. This project highlights the interest of biomimetic artificial muscle technology. Indeed, the major

advantage of this robot is that it is driven by pneumatic artificial muscles. These muscles behave in a manner similar to that of human muscles. The use of such actuators is an innovative concept in the field. This technology gives flexibility to the movements of the robotic arm and allows the robot to be controlled in stiffness. Rehabilitation will benefit from this last point as the difficulty of the robot's movement can be adjusted according to the patient's abilities.

After the robot has been conceptualised and designed, it must be put into operation and made usable. This is the main goal of this thesis. A first objective was to model and validate the very specific physics of the actuators. In order to do this, an experimental model was developed, specifically for the muscles used. Based on the model, the limits and behaviours of the robot were characterised. Then, having a good knowledge of the robot's functioning, the control of the robot could be designed. It was implemented step by step: first via low-level strategies to control the position, force or stiffness felt at the handle, and then via a high-level strategy that highlights a human-robot interaction task. The different steps of this work were accompanied throughout by an experimental approach, which allowed to validate the model and the control of the robot.

The report is organised as follows: The *first chapter* gives a state of the art of the fields that fall within the scope of the thesis, such as neurorehabilitation, robots used in rehabilitation therapies and artificial pneumatic muscles. The *second chapter* presents the robotic arm used. It describes the robot, its components and its complete system. It also situates it in its reference space, at its starting position. The *third chapter* investigates the model of the muscles that is implemented in the system. Several approaches are described and a final model is selected. The *fourth chapter* proposes an analysis of the limits of the system. The maximum force and stiffness achievable by the robotic arm in the entire workspace are determined. This provides the user with basic information for the conceptualisation of games that can be performed with the robot. The *fifth chapter* deals with the low-level control of position, force or stiffness of the system. It contains the description, implementation and validation of this control. The *sixth chapter* implements a high-level task in order to obtain a typical human-robot interaction. The task focuses on a hybrid stiffness control to make the interface compliant. Finally, the *seventh chapter* proposes a conclusion to the work, as well as a list of mechanical improvements to be made to the arm design. A personal perspective concludes this thesis.

Chapter 1

State of the Art

The aim of this chapter is to present the current state of knowledge in the various fields related to the subject of this thesis. Given the intrinsic link between robotics and the medical world in the context of this project, a first part of the section discusses the consequences of a stroke and the importance of robotics in the field of rehabilitation. A description of neurorehabilitation is also presented. Next, an overview of some of the robots that have been developed to date for rehabilitation therapy is provided. This chapter then introduces pneumatic artificial muscles (PAMs), their functioning, and their characteristics. Finally, an overview of PAMs in practice is provided, discussing their challenges, goals, successes and failures.

1.1 Neurorehabilitation

Neurological conditions, such as traumatic brain injury, will impact our quality of life by impairing our brain function. While many diseases have their own challenges that need to be addressed in clinical care they do share common basic principles in their course of recovery that we can exploit with a shared similar treatment plan. Stroke is the most common neurological condition found in the hospital and in clinical care. The stroke and the incidence of a stroke can impair motor sensory or cognitive tasks. The symptoms caused by stroke range from mobility problems to visual speech and communication, difficulties in swallowing, neuropathic pain and emotional difficulties such as anxiety and depression. In 80% of all stroke survivors, motor function is deeply impacted by the incident of the stroke, with symptoms including loss of balance, difficulties in walking and grasping objects, trouble to recognize the surrounding, muscle fatigue and lack of coordination. Patients can not go back to their normal lives right away.

A stroke is caused by a sudden reduction of blood flow to areas of the brain which can either be caused by local bleeding or by a blood clot that blocks the blood flow. Both of

these incidents will cause a reduction of oxygen supply to the downstream neuronal areas causing the neurons to starve to death because they are missing their nutrition. A massive increase in cell death occurs in the hours after stroke, this is called the acute phase. Days to months after the initial injury, newer regenerative processes start to happen, this is called the chronic phase. Treatment options have to be developed for both phases individually.

During the *acute phase*, the best course of action here is to re-establish blood flow as fast as possible so that the neurons who are slowly starving will get their nutrition back. The first way to achieve this is to apply a thrombolytic therapy using tPA which is an enzyme that will eat away the blood clot. An additional approach is to supplement food for the neurons by using pharmaceutical treatment. However the long term studies are still missing. The goal of either treatment is to be applied as early as possible to save as much brain as possible. During this phase, time is key. Yet, only 5% to 10% of all stroke patients are actually eligible to early treatment [20].

During the *chronic phase*, neurorehabilitation comes into play.

> *Neurorehabilitation is a clinical subspecialty devoted to the restoration and maximization of function that have been lost due to impairments caused by injury or disease of the nervous system [20].*

The main goal of neurorehabilitation is to help the patient to improve their functions for activities of daily living. It aims to accelerate recovery, reduce symptoms, elevate the well-being of the patient, and maximise his independence. After a stroke, patients recover to some extent without the benefit of treatment. However, they do not return to the basic level; there is room for improvement. Spontaneous recovery can be taken one step further by better understanding neuroplasticity.

> *Neuroplasticity describes the mechanisms by which the brain is capable of modifying itself during neurogenesis processes as early as the embryonic phase or during learning [34].*

Physical exercise leads to an increase in synaptic strength and a general improvement in brain health. This strengthens the circuits of specific movements that ultimately leads to better behaviour. Neurorehabilitation can be considered as applied neuroplasticity. It uses the brain's ability to learn how to further strengthen weakened circuits or create new ones to replace as many functions as possible. Neuroplasticity is then the way to reorganize neural pathways as a result of experience or exercise. There are two types of plasticity in the context of functional recovery:

1. *Functional plasticity*: the brain's ability to move functions from a damaged area of the brain to other undamaged areas (Fig. 1.1).
2. *Structural plasticity*: the brain's ability to actually change its physical structure by growing axons and synapses as a result of learning (Fig. 1.2).

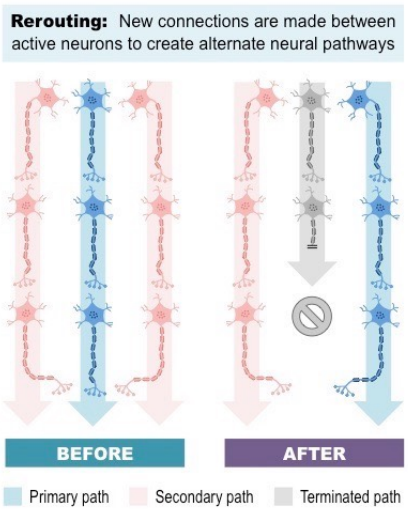


Figure 1.1: Functional plasticity [8].

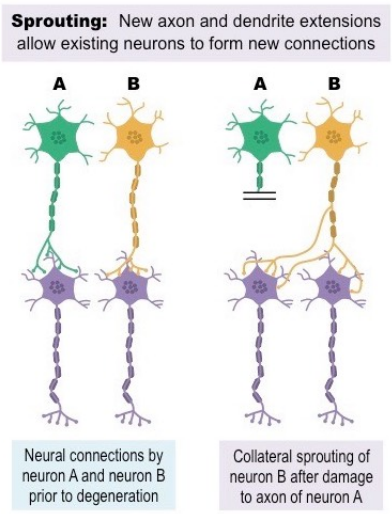


Figure 1.2: Structural plasticity [8].

There are 10 principles of neuroplasticity [20]:

1. *Use it or lose it*: Repetition and practice strengthens neural connections while connections in the brain that are not used become weak.
2. *Use it and improve it*: High intensity training can further induce plasticity to increase functional recovery.
3. *Specificity*: Training of a specific task may only induce changes in the underlying circuitry and my not adapt other circuitries.
4. *Repetition matters*: Repetition is necessary to consolidate the newly learned/relearned behaviour. Enriched environment leads to increased neuroplasticity and better recovery as it stimulates to repeat specific tasks.
5. *Intensity matters*: Intensity matters to facilitate learning of newly/re-learned behaviours.
6. *Time matters*: The brain is in a special state to learn right after injury. There is a critical period of rehabilitation that provides a specific time window during which treatment options might be particularly effective. Delayed treatment might result in compensatory movements that are maladaptive.
7. *Salience matters*: The training experience must be specific, particularly salient to induce plasticity.
8. *Age matters*: Neuroplastic responses are altered in the aged brain. Synaptogenesis and cortical map reorganisations are reduced.
9. *Transference*: The ability of plasticity within one set of neural circuits to promote concurrent or subsequent plasticity. System is already in an excitable stage and thus is more open to consolidate newly/re-learned behaviour.

10. *Interference*: The ability of plasticity to impede the induction of new plasticity within that same circuitry. System is not in a state that allows learning.

1.2 Types of robots used in assisted therapy throughout history

Robotic arms are an important research area today. They allow linking the mechanics of an external skeleton with the intelligence of a human being. The development of the first medical robot for rehabilitation dates back several decades [4]. Upper limb rehabilitation robots can be divided into two classes: exoskeletons and end-effector robots.

The ***exoskeletons*** impose their articular movements on the entire limb of the patient. The *ArmeoPower* robot (Fig. 1.3) [15] is one example. This device is a robotic arm interface for upper limb rehabilitation, which has been developed and is marketed by the company *Hocoma*, specialised in robotic rehabilitation therapy for neurological movement disorders. The 6 DoFs actuated allow training in a large 3D workspace. The guidance of the arm movements can be adapted to the needs of each patient. It is adjusted according to the subject's progress and changing abilities during therapy. It is very helpful for patients who are not very active, or acts transparently for those who do not need help. This variable assistance motivates patients to actively participate in their training.

Figure 1.3: ArmeoPower robot.

The ***end-effector robots*** bind the patient only by the terminal organ. The most recent example is the *REAplan* robot (Fig. 1.4) [3], developed by a spin-off of UCLouvain and marketed by the Belgian company *Axinesis*. This medical device is an upper-limp end-effector, equipped with a heigh-adjustable table, force sensors, motors, a screen and a speaker to interact with the therapist.

Figure 1.4: REAplan robot.

Other prototypes of all kinds have existed through time and have enabled the current status quo. Among the ancestors of the robots presented above, there is the *MIT-MANUS* robot (Fig. 1.5) [26] [14]. This device has a manipulator with 2 DoFs. Like the REAplan, it moves the hand of the patient in an horizontal plane. This robot can assist or resist the subject. There is visual, audio or tactile feedback creating passive or active interaction between the patient and the robot. The robot is programmed to interact with the patient with low impedance. This gives it a soft feel during movement. Therefore, the interaction between the robot and the patient is safe, stable and compliant. Other examples are the *ARM-GUIDE* robot (Fig. 1.6) [14] or the *MIME* robot (Fig. 1.7) [26] [14]. The first one uses a 3D workspace where the arm slides along a rail which gives 6 DoFs. It can also assist or resist the subject. It contains sensors for position, speed and force. It is used both as a diagnostic tool and a treatment tool. The second robot interacts stably and repeatedly with the patient. A force is applied by the manipulator robot during the movement towards the goal. It contains a force-torque sensor to measure the force interaction between the robot and the subject. The patient performs the movement in front of a mirror.

Figure 1.5: MIT-MANUS (Interactive Motion Technologies).

Figure 1.6: ARM-GUIDE (Assisted Rehabilitation and Measurement Guide).

Figure 1.7: MIME (Mirror Image Movement Enabler).

Bringing robots into the medical world represents a significant step forward in rehabilitation. However, it has certain disadvantages. The most important is the safety of the interaction between the device and the human. In addition, this equipment must be accepted by the medical body. It is important to remember that the therapist's eyes and hands are always helpful, the robot quantifies and provides the data. Also, some skills are required to work with robots. Therapists must be trained to use the equipment safely and need sometimes to understand both software and hardware to perform any necessary set-up [14]. These robot-assisted therapies are also carried out individually, not in groups, which is less cost-effective in terms of time and can affect patient motivation. Despite these few drawbacks, the field of robot-assisted therapy for stroke rehabilitation has a future.

1.3 Pneumatic Artificial Muscles (PAMs)

One type of actuator for medical robots are pneumatic artificial muscles.

1.3.1 Description

PAMs were first developed under the name of *McKibben Artificial Muscles* in the 1950s [2]. They are constructed with an inner rubber tube covered with an outer layer of braided nylon (Fig. 1.8) [5]. Those muscles are closed by air input on one side and the force attachment point on the other. The elementary pantograph network formed by its helical weaving allows an energy transfer [32]. When a certain pressure P is applied, circumferential stress exerted on the cylinder walls is converted into axial contraction force F. This action increases the radius, decreases the length of the muscle, and therefore allows the muscle to contract. The light and compact cylindrical shape of the muscle allows it to develop a great maximum strength. By changing the pressure, the stiffness[1] of the muscle changes [11].

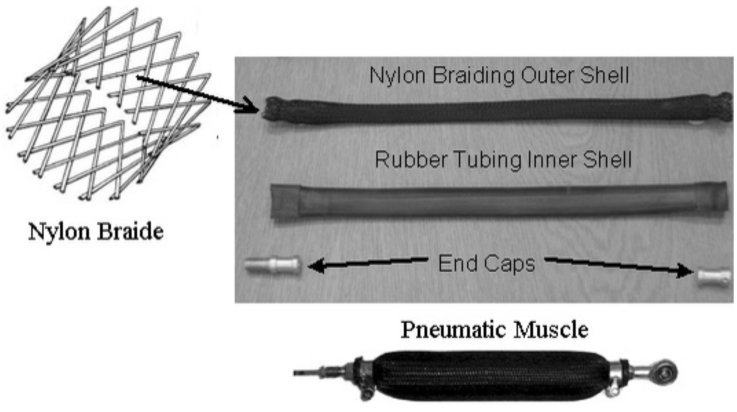


Figure 1.8: PAM design [5].

[1] *Stiffness* is the characteristic that indicates the resistance to elastic deformation of a body.

The actuator behaves like a spiral spring and has a natural stiffness. The McKibben muscle actuator gives the robot joint a passive compliance[2], called *natural compliance* [32]. The latter depends on the material used to build this artificial muscle and is due to complex elastic and friction phenomena [31]. This natural damped spring gives an analogous behaviour to the natural skeletal muscle. Therefore, McKibben artificial muscles allow to reproduce the human softness joint.

PAMs respect the following principle: if a robot-arm - kinematically analogous to the human arm - is powered by actuators - whose stiffness is similar to that of the human joints - then the compliance at the end of the robot-arm will match that of the human arm [31].

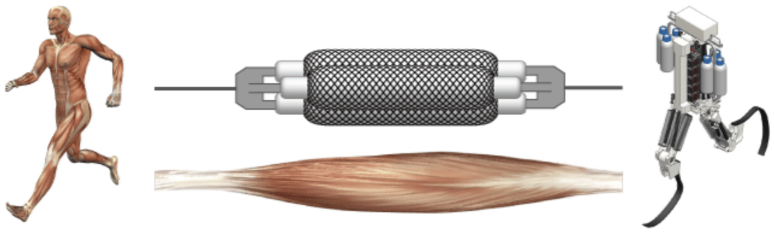


Figure 1.9: Analogue behaviour between PAMs and human muscles [27].

The contraction of the McKibben muscle is naturally damped by nonlinear kinetic friction inherent to its braided shell [32]. This creates a paradox which is one limitation of the McKibben muscle actuator: The dry friction muscle disturbs the torque in the joints. The dynamic accuracy when speed increases is also deteriorated by friction. There is a loss of accuracy when changing from dry to kinetic friction. However, this kinematic friction produced by the muscle thread results in natural compliance. There is therefore a compromise in terms of the presence of friction.

1.3.2 Control

Good modeling and control of PAMs is essential to maintain their *soft* properties. These systems use controlled pairs of PAMs, configured antagonistically, as power sources [5]. Pneumatic actuators drive cables and pulleys, via revolute joints. Basically, when one muscle contracts, the other one expands. As shown in Fig. 1.10, this produces an angle and therefore antagonistic torques in the joint. This allows joint movement. Two elements are necessary to have bidirectional movements and forces.

[2] *Compliance* is the ability to adapt to an external disturbance.

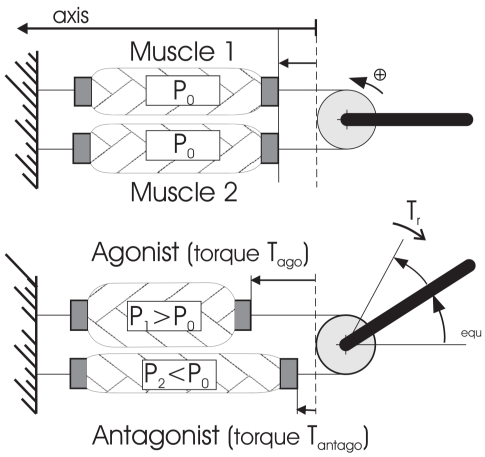


Figure 1.10: Working principle of the PAMs [31].

The electrical system required for this exoskeletal device consists of I/P converters, sensors and a robot controller [31]. Control pressure is the input, muscle length is the output. The McKibben muscle is controlled by servo-valves that supply the system. Usually, the pressure is regulated by a PID controller which decides the time of opening and closing of the valves [5]. An force-torque sensor is used to detect the desired movement of the user and to define the force to be applied to it.

1.3.3 Force

In the literature of McKibben's muscles, there is the expression of the force F developed by each muscle as a function of the control pressure P and the contraction ratio ε [32] [31].

$$F(\varepsilon, P) = (\pi r_0^2)\ P\ \left[a(1-\varepsilon)^2 - b\right] \qquad \text{with} \qquad \begin{cases} \varepsilon &= \frac{(l_0 - l)}{l_0} \\ a &= \frac{3}{\tan^2(\alpha_0)} \\ b &= \frac{1}{\sin^2(\alpha_0)} \end{cases}$$

where l_0 is the initial length of the muscle, α_0 is the initial muscle braid angle, and r_0 is the initial muscle radius. The mathematical development to find this model is given in Appendix A.

It can be observed that the static force is globally proportional to the cross-sectionnal muscle area (πr_0^2) and the control pressure P. The initial muscle braid angle α_0 and the initial muscle radius r_0 are two parameters that can be dimensioned to define the maximum force generated by the muscle. When α_0 decreases, the maximum static force increases. The smaller α_0 is, the more the muscle radius increases during contraction. This action is limited by the type of rubber used. It must tolerate the possible elongation. The power of the McKibben muscle is also limited by the dynamic behaviour of the servo-valve. A

typical value of maximum muscle power during contraction is 75 W, which is quite low [31]. Increasing this value would be possible by increasing the bandwidth of the servo-valves. The low response time of the McKibben muscle comes from this low bandwidth. This artificial muscle is the slowest, although it is the strongest.

1.3.4 Torque & Joint range

The torque developed at the joint is [32] [31]:

$$\tau = R \cdot (F_{Muscle2}(\varepsilon_2, P_2) - F_{Muscle1}(\varepsilon_1, P_1))$$

The actuator joint range is [32] [31]:

$$\theta_{range} = \left[-\frac{\varepsilon_0 \ l_0}{R}; +\frac{\varepsilon_0 \ l_0}{R} \right] \ (rad)$$

where l_0 is the initial length of the McKibben muscle actuator, ε_0 is the contraction ratio of the muscle in the initial state, and R is the radius of the pulley.

The joint range depends on the initial length of the McKibben muscle actuator, l_0. The design of the parameter R offers a compromise between the maximum torque of the actuator and its joint range. To achieve high angular joint ranges, a high value of l_0 is required. This relationship between the joint range and l_0 is the major negative point of pneumatic muscle actuators. A good trade-off must therefore be found between the acceptable dimensions of the robot and the desired dynamic performance [31]. Also, the parameters α_0 and r_0 can be used to modify the maximum force and torque without influencing the joint range.

The joints must be controlled in a robust manner to withstand the dynamic variations of the other joints [32]. The joint positions of the robot can be controlled in open loop - because the antagonistic muscle actuator is stable - or closed loop - with a simple linear PID controller [31]. The joints are influenced by the dynamic effects of others, as well as by gravity.

1.3.5 Advantages

Actuation systems provide the robot with forces, torques and mechanical motions. Their performance is judged according to several parameters such as power, strength, response rate, physical size, speed of motion, reliability, controllability, compliance and cost [5]. Pneumatic muscle actuators are excellent candidates and meet most of the above criteria. They have inherent controllable compliance which is beneficial for safety. Indeed, this compliant behavior permits a soft and safer interaction with the user. PAMs have an excellent power/weight ratio. They have many other advantages such as lightness, gravity compensation, ease of fitting and adjustment, relatively low mechanical complexity and stability. Finally, they are a low cost solution.

1.4 PAMs in practice

Several prototypes powered by artificial pneumatic muscles have already been developed. Taking into account their pros and cons, they must meet certain objectives.

1.4.1 Challenges & Objectives

An important issue is the acceptance of the robot by the patient. Indeed, current rehabilitation robotics is patient-centred. The rehabilitation robot must be designed according to the subject's feelings and his handicap, in order to involve the patient in this mechanized rehabilitation. To achieve this objective, the design was created to be as light as possible. The artificial muscles provide stiffness comparable to human joints. Pneumatic silencers are placed on the artificial muscles. They are relatively noisy due to the escape of pressurised air with each contraction, and this can affect the patient. Given the importance of the patient's feeling in rehabilitation, the motivation of patients with the clinical use of rehabilitation robots must also be evaluated. Their comfort, anxiety and tolerance are criteria that must also be taken into account [14]. It would be judicious to compare the patient's perception according to whether the rehabilitation is done by a robot with an electric motor or a robot with artificial muscles.

The most important point is the safety of the interaction between the device and the human [5]. Safety is achieved by combining high strength with actuator and structural compliance. It is necessary to guarantee the safety of the entire rigid mechanical structure. This safety should be ensured by the presence of a force sensor and the use of easily adjustable control pressures.

Another challenge of the PAMs is to provide adapted and progressive assistance for each patient. The robot must be able to support the patient in proportion to his needs. It should even create resistance and act as an obstacle to movement in order to improve progress. To do so, the robot can adopt an adaptive assistance strategy. This means that the force that assists the patient decreases according to the patient's performance, to make the task more difficult if the challenge is not high enough. This is achieved through impedance control [29]. Special care must be taken not to create a slackening of the patient. Indeed, slacking hypothesis states that, if the patient receives too much assistance, he does not provide the maximum of his skills because he relies on the robot's help [24]. This is obviously to be avoided so that rehabilitation can be at its most effective.

Several studies show the importance of integrating rhythmic movements into rehabilitation therapy in order to regain full motor recovery [23] [25]. Rhythmic upper-limb movement is a different motor primitive than discrete one. They are controlled by distinct neural circuitries and require different cortical networks. The rhythmic movement is generally less damaged by stroke [23]. The combination of the two is useful for many everyday movements. This is why rehabilitation must target both movements. Rhythmic movements are simple to

perform. There is even a slight learning transfer from discrete to rhythmic movements, but none in the other direction [25]. An essential objective of the use of robots in therapy is the rehabilitation of various rhythmic movements, which are easy to implement with a robot. This aims to promote a faster recovery of discrete upper limb movements that are essential to daily life. The robot activated by artificial muscles would benefit from the positive effect of a rehabilitation which favours certain discrete and rhythmic gestures.

1.4.2 Successes & Failures

MAPs have been around for some time. Their use is attractive for several fields such as medical and biorobotic applications of course, but also for industrial or even aerospace applications [2]. However, these robotic systems have often been developed for research purposes. These actuators have not been widely used in practice so far. A few examples can be cited:

- In 2003, the Department of Orthopaedic Surgery Prince of Songkla University (Thaïland) developed a PAM-actuated forearm prosthesis for a patient who lost his arms (Fig. 1.11) [36]. The movement of the artificial arm is manually controlled by a foot pedal. This prototype has been clinically tested but has not been commercialized.
- In 2004, H. Kobayashi and K. Hiramatsu proposed an armor-type muscle combination that provided muscle support for paralyzed patients (Fig. 1.12) [21]. Further research is needed to enable the robot to perform the full range of upper limb movements. A test system for a commercial product was envisaged, but does not seem to have been validated.
- In 2005, RUPERT is developed by the Arizona State University (Fig. 1.11) [12]. It is a therapeutic device powered by four PAMs with 5 DoFs. This device has disadvantages in terms of control, such as response times that are too slow or unsatisfactory precision. Control algorithms are under development to meet therapeutic criteria. But again, there is no sign of this product on the market today.

Figure 1.11: Forearm prosthesis developed by the Songkla University.

Figure 1.12: Armor-type muscle combination proposed by two Japanese.

Figure 1.13: RUPERT developed by the Arizona State University.

These robotic systems tend to be developed for research purposes, being too complex for practical use. The fact that a commercial phase is not often reached is not easily explained. One of the reasons is the lack of large-scale need for this specific type of actuator. Therefore, there is a lack of technological effort to improve existing designs [6]. Other reasons come from the disadvantages of PAMs. First, control of these muscles is not trivial. In addition, its longevity can be a disadvantage. It varies from 100,000 to 1,000,000 cycles. This limitation is due to the stress concentrations between the flexible membrane and the rigid tips. Then, the existing pneumatic muscles do not allow a deformation greater than 35% of their initial length, which can limit the range of operation. Additionally, the noise generated by the pressure also remains a major problem, which can make the patient reluctant to use this type of device. Nevertheless, as previously seen, PAMs have two great qualities in comparison to the electrical control that have made them successful and which give them the merit of being exploited.

- *Compliance*: Conventional electric motorisation induces a great rigidity of the robot. This imposes a mechanical impedance on the patient's arm during his rehabilitation movements which may be too high. Pneumatic artificial muscles, with their strong static and dynamic analogy to the skeletal muscle, can provide the natural compliance that is lacking in electrically powered rehabilitation robots.
- *Strength & Lightness*: The structure of a robot arm composed of PAMs is lighter than electrical drives. This results in a much higher payload to weight ratio and an advantageous power/weight ratio.

1.5 Summary

Stroke is a disease that affects a large number of older people, causing physical deficiencies but also psychological and social damage. Treatment should be carried out as soon as possible in order to recover maximum motor function. The intervention of robots in the therapy improves the progress of the patients, as the treatment must contain intense and repetitive movements. A new type of actuator is gradually entering the field of medical robotics: pneumatic artificial muscles. These are elastic tubes which, when supplied with pressure, swell and contract. When two actuators are arranged antagonistically, they create movement and joint torque. The major advantage of these actuators is that they are naturally compliant. On the other hand, the control becomes rather imprecise. There is no trace of such actuators in commercial medical robots yet. It is therefore interesting to develop a prototype of a robot driven by pneumatic artificial muscles in order to contribute to the rehabilitation of people who have suffered a stroke. This project is the subject of this master thesis. The next chapter describes the robotic arm developed for this study.

Chapter 2

Description of the robotic arm and its system

This chapter introduces the robot arm, as well as the system as a whole used in the project. The components that constitute the robotic arm and the system are described. It also places the robot in its frame of reference in order to define the notations used in the following chapters. Finally, a brief explanation of how the robot is placed in its starting position is given.

2.1 The robotic arm and its system

The 3D model of the robot arm is presented in Fig. 2.1. The robot is based on the SCARA model. It has 2 DoFs: *flexion-extension* of the shoulder, *flexion-extension* of the elbow. It is operated by 4 pneumatic artificial muscles. Via a system of cables and pulleys, the two lower muscles control the movement of the shoulder, while the two upper muscles control the movement of the elbow.

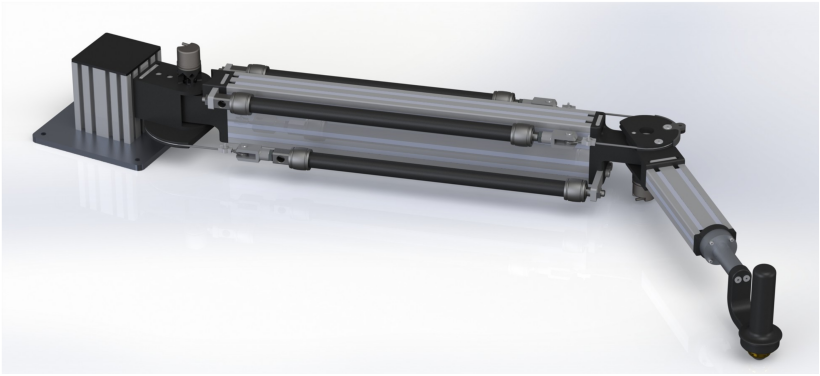


Figure 2.1: 3D model of the robotic arm.

The robot is integrated into a complete system, shown in FIG. 2.2. The system works as follows: The control is done via the LabView software. A current command is sent. This is converted into a pressure command by using Intensity/Pressure converters and the compressed air supplied to the robot. The pressures selected for each muscle are sent to the robot, causing it to move. The two angle sensors and the force sensor - the system's outputs - send their data to LabView. A control is therefore possible, in order to adjust the values of the pressures to be sent to the muscles, according to the data received by the sensors. The link between the outside world and the software is made through a CompactRIO, a control unit manufactured by National Instruments.

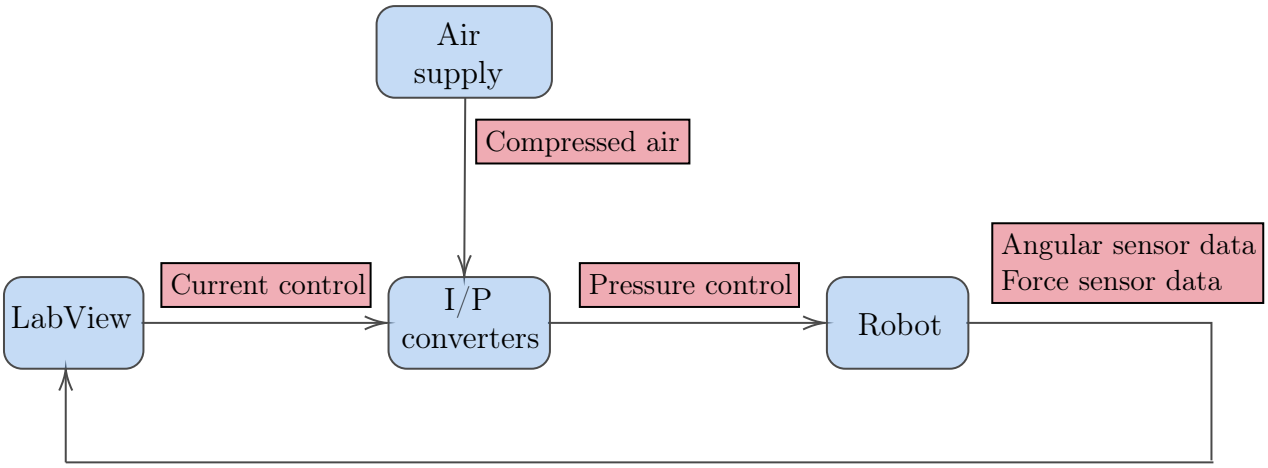


Figure 2.2: Block diagram of system operation.

THE FRAME

The frame is the fixed reference. It must be rigid in order to support the whole structure and to reduce the deformation due to bending. It is a machined part, constructed of aluminium. It is a square shape, measuring 100x100 mm with an 8 mm wide slot. It is fixed to the table with four screws placed at each corner of an aluminium plate.

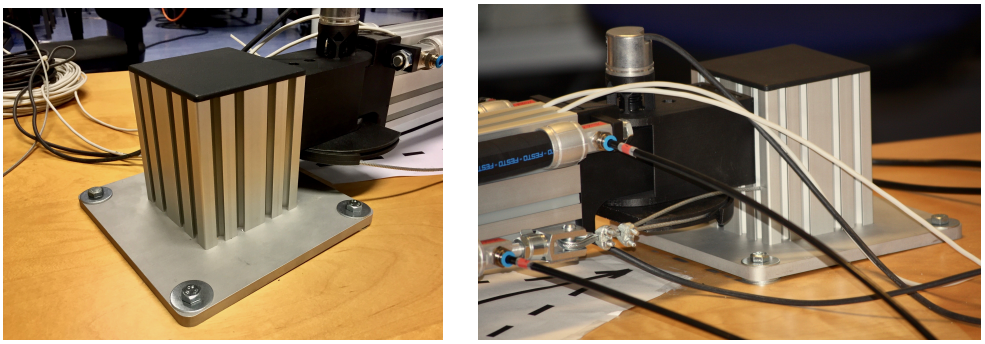


Figure 2.3: The frame.

The segments

Two solid segments represent the human arm and forearm. They are designed by commercially available aluminium profiles. They contain slides that guide the cables and facilitate the attachment of the artificial muscles, while reducing the weight and cost of the robot. They are a rectangular shape. The first segment is 50x100 mm, with an 8 mm wide slot, and a length of 700 mm. The second one is 30x60 mm, with an 8 mm wide slot, and a length of 450 mm.

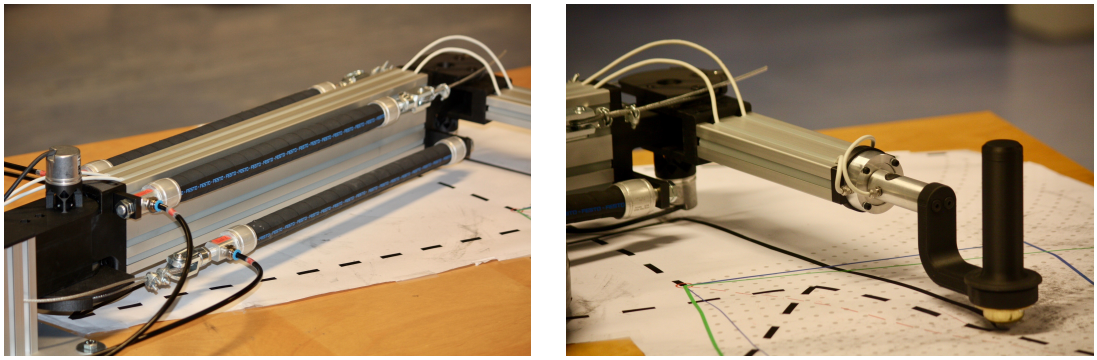


Figure 2.4: The segments.

The joints

Two joints connect the solid parts to each other via a rotating system. The pulleys have two pins to allow the attachment of the muscle cables. These simple parts are printed in 3D. The material used for 3D printing is onyx and kevlar.

Figure 2.5: The joints.

The handle

A handle provides the link between the robot and the user. It is mounted on a non-motorised pivot, allowing the user to grasp the robot freely. This piece is 3D printed with onyx. The handle rests on a carrier ball, which limits the friction between the handle and the contact surface, here the table.

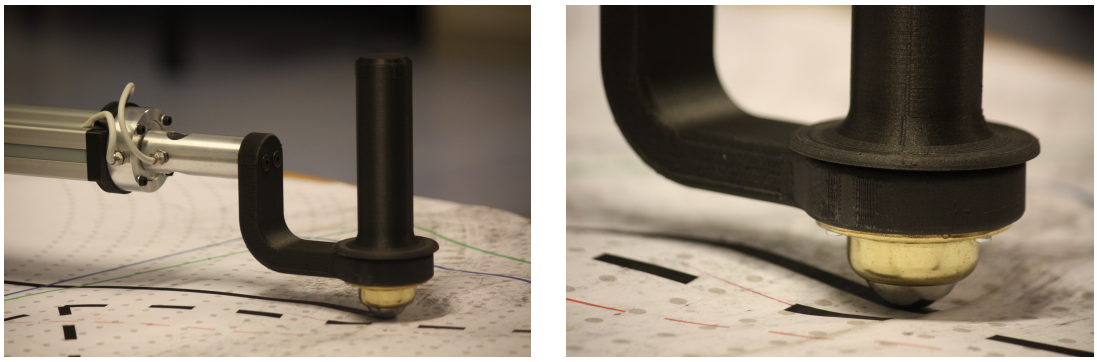


Figure 2.6: The handle.

The cables

The muscles of a joint are connected by means of metal cables that run through the pulleys. These are galvanised steel with 7 strands and a diameter of 3 mm. They are fixed with cable ties and steel wire thimbles and form an arc around the pulley.

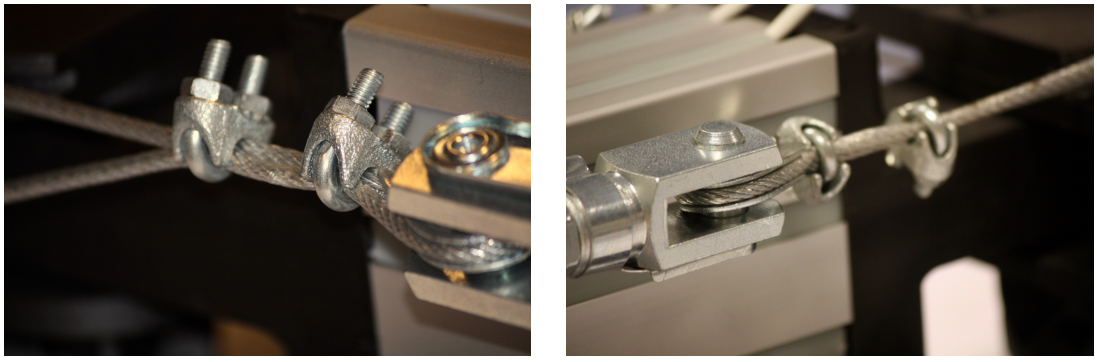


Figure 2.7: The cable.

The pneumatic artificial muscles

Four pneumatic artificial muscles are located on the arm segment. The muscles used are those of Festo. They are 400 mm long and have a diameter of 20 mm at rest. They are equipped with a compressed air supply that allows the contraction of these muscles and thus the movement of the robot.

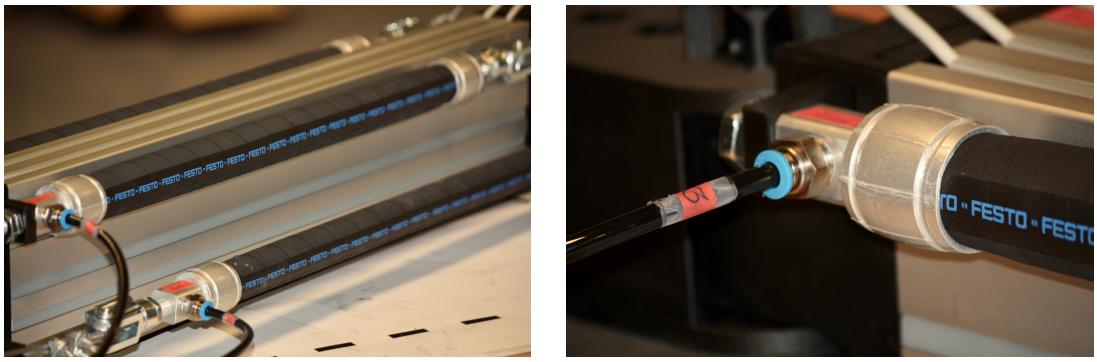


Figure 2.8: The pneumatic artificial muscles.

The angular sensors

Two angular sensors are placed at each joint. They track the movement of the arm and forearm. They are used as output information from the system. The rotary encoders used are Hedenhain ROD 1020s. They are characterised by their rugged design and compact dimensions. Their external diameter is 36.5 mm. It provides a 5V TTL quadrature pulse output signal. The resolution, measured in pulses per revolution, also known as line count, is from 100 to 3600 [13].

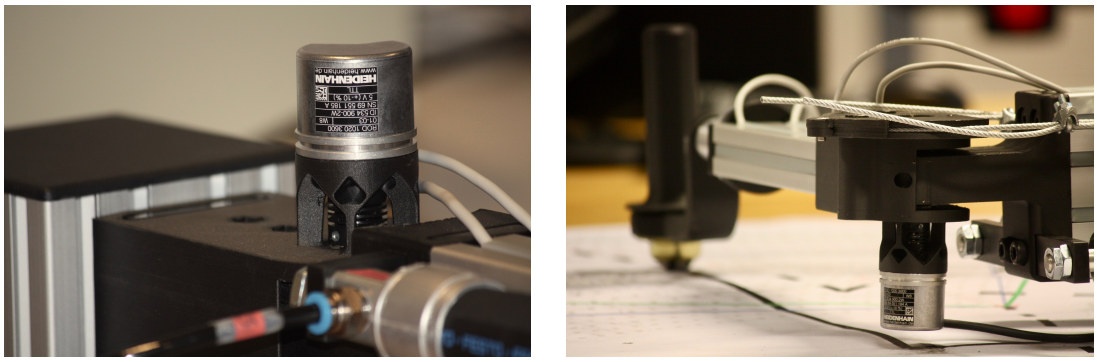


Figure 2.9: The angular sensors.

The force sensor

The system is equipped with a force sensor, located at the level of the handle, to control the force of the interaction between the user and the device. The force sensor used is the Scaime model FDC 20-20kg-SP/FME1944. The load rating according to x and z is 20 kg. It has a sensitivity of $0.2\,\mathrm{mV\,V^{-1}}$ and $0.7\,\mathrm{mV\,V^{-1}}$ along x and z respectively. The z-axis is defined along the length of the sensor, the x-axis is along the radius of the sensor.

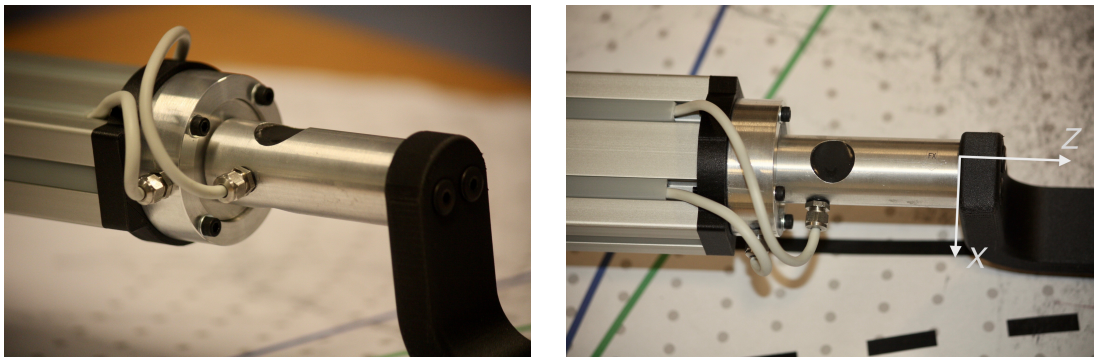
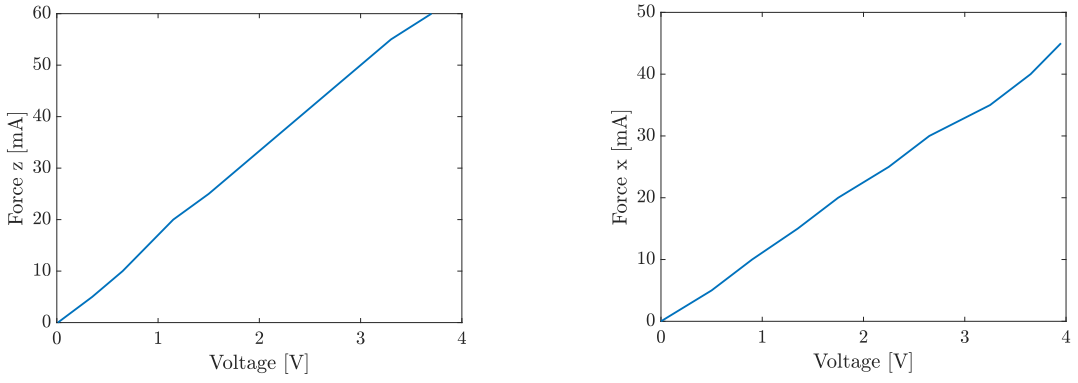


Figure 2.10: The force sensor.

The force sensor values are used as system output information. The intensity of this force is measured in voltage by the sensor. A relationship between the value of the output voltage from the sensor and the force it represents has been calculated experimentally. The procedure is as follows: A force is gradually applied to the handle in z direction - direction extended from the forearm -, and in x direction - direction perpendicular to the forearm and parallel to the arm. The value of the applied force is measured with a dynamometer. The value of the corresponding voltage is measured with a multimeter. The experiment provides the force-voltage relationship illustrated in Fig. 2.11.



(a) z-axis

(b) x-axis

Figure 2.11: Relationship between the output voltage of the sensor and the measured force.

The relationships are close to linear and can be approximated by the following equations:

$$F_z\ [N] = \frac{1}{0.06} \cdot U_z\ [V]$$
$$F_x\ [N] = \frac{1}{0.09} \cdot U_x\ [V]$$

with U_z and U_x, the voltages corresponding to the forces in z and x respectively.

The Intensity/Pressure converters

The system is composed of Intensity to Pressure (I/P) converters which, as the name suggests, convert the intensity - sent in Ampere via the software - into a certain pressure - sent to each muscle of the robot. The pressure is transported from the box to the muscles via plastic tubes.

Figure 2.12: The Intensity/Pressure converters and the tubes.

The devices convert a range of [0.4 - 20] mA into [0.1 - 5] bar. This linear relationship has been verified experimentally using a pressure reader. The procedure is as follows: For each pressure (P), from 0 to 5 bar, a current value (I) is calculated based on the linear relationship of the converters. This value is sent to the system. A pressure reader is connected in series between the I/P converters and a robot muscle. Depending on the pressure that the pressure reader receives as input, it returns a certain current. This current value is displayed on a multimeter. This operation is repeated for each muscle. The Tab. 2.1 compares the current values sent via the software LabView and those displayed on the multimeter for each muscle. The values are shown in Fig. 2.13.

		I read on the multimeter [mA]			
P [bar]	**I from LabView [mA]**	**Muscle 1**	**Muscle 2**	**Muscle 3**	**Muscle 4**
0	3.67	4.23	4.23	4.16	4.15
1	6.94	6.58	6.31	6.36	6.37
2	10.20	9.99	9.62	9.66	9.65
3	13.47	13.20	12.86	12.87	12.85
4	16.73	16.48	16.38	16.38	16.35
5	20.00	20.00	19.72	19.70	19.65

Table 2.1: Comparison of the current values sent via the software (*I from LabView*) and those displayed on the multimeter for each muscle.

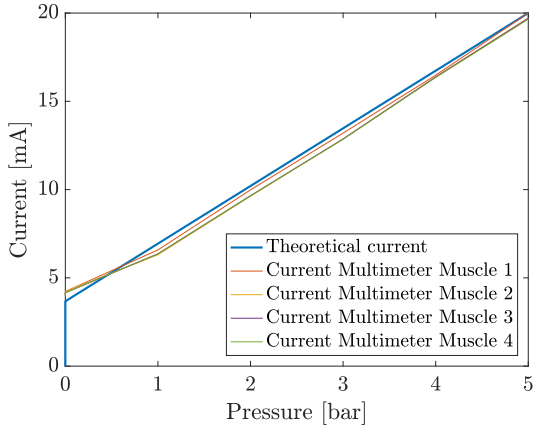


Figure 2.13: Comparison of the current values sent via the software (*Theoretical current*) and those displayed on the multimeter for each muscle.

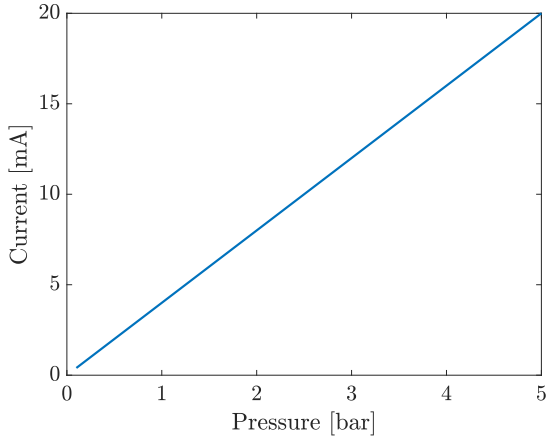


Figure 2.14: Linear relationship between current and pressure of I/P converters comparison.

The current values are similar, with an accuracy of half a mA. This confirms the reliability of the I/P converters used. The linear relationship between currents and pressure, shown on the fig 2.14, is awarded.

The relationship has the following equations:

$$P\ [bar] = 326.67 \cdot I\ [A] - 1.63$$

The CompactRIO

The CompactRIO is a device from the National Instrument industry that provides an industrial control and monitoring solution using sensor or protocol-specific packaged I/O modules with real-time capabilities [19]. It is used to realise the communication between the sensor data, the FPGA and the CPU. It is composed of different modules that respectively take care of reading or sending information through the system.

- Module 1 is the NI 9265. It is a 4-channel, 16-Bit analog current output module. It takes care of sending pressure commands from the software to the robot through the FPGA interface [16].
- Module 2 is the NI 9403. It is a 32-channel, TTL digital input/output module. It is used to receive information from the angular sensors [17].
- Module 3 is the NI 9215. It is a 4-channel, 16-Bit simultaneous analog input module. It is used to transmit the load cell information through the FPGA interface [18].

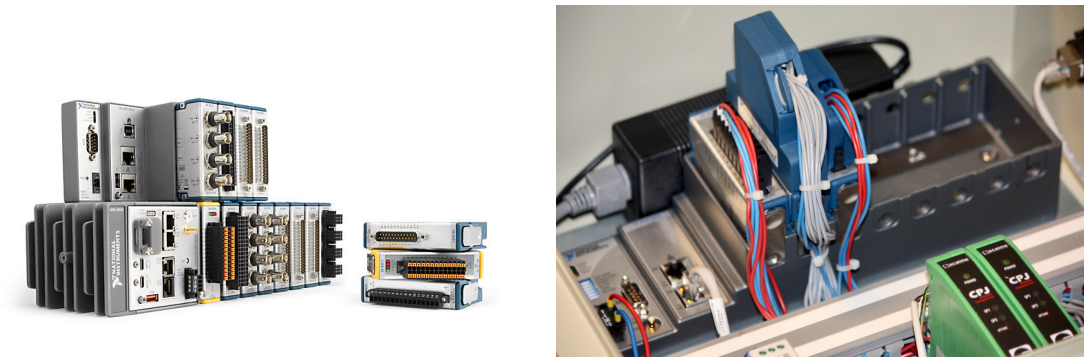


Figure 2.15: The CompactRIO and its modules.

2.2 The robot in its reference frame

The main Cartesian reference is fixed to the frame. In the neutral position, the arm is on the y_0-axis and the forearm is parallel to the x_0-axis. The θ angles respect the kinematics used in the SCARA kinematic models. The α angles represent the movements of the arm and forearm around their initial position. The relationships between the different angles are as follows:

$$\begin{cases} \theta_1 &= 90^\circ + \alpha_1 \\ \theta_2 &= -90^\circ + \alpha_2 \end{cases}$$

with

$$\begin{cases} \alpha_1 & = [-22.5°; 22.5°] \\ \alpha_2 & = [-32.5°; 32.5°] \end{cases}$$

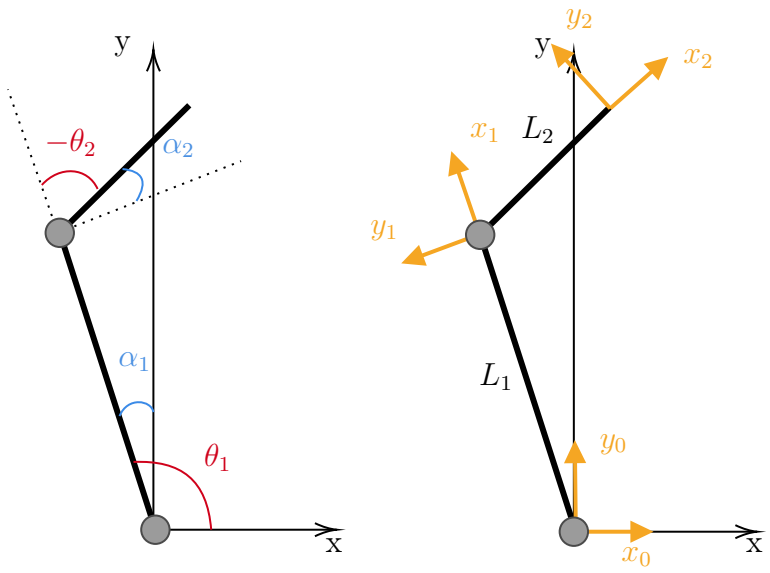


Figure 2.16: Robotic arm in its coordinate system.

Changes in reference frame follow the following relationships:

$$\begin{cases} x_0 & = 0 \\ y_0 & = 0 \end{cases} \qquad \begin{cases} x_1 & = x_0 - L_1 \cdot \cos(\alpha_1) \\ y_1 & = y_0 + L_2 \cdot \sin(\alpha_1) \end{cases}$$

$$\begin{cases} x_2 & = x_1 + L_1 \cdot \cos(\alpha_1 + \alpha_2) = -L_1 \cdot \cos(\alpha_1) + L_1 \cdot \cos(\alpha_1 + \alpha_2) \\ y_2 & = y_1 + L_2 \cdot \sin(\alpha_1 + \alpha_2) = L_2 \cdot \sin(\alpha_1) + L_2 \cdot \sin(\alpha_1 + \alpha_2) \end{cases}$$

2.3 The robot in its starting position

Each time the program is started, the robot must be placed in its starting position, i.e. at a 90° angle at the elbow. During the exercise, the arm is able to reach a certain space, shown as dotted lines in Fig. 2.17. However, for safety and ergonomic reasons, the final working space used during the sessions is shown as a blue rectangle. It is a rectangle with dimensions of 459x305 mm, whose base is parallel to the edge of the table. The important dimensions of the robot are listed in Tab. 2.2. For each joint, the length of the segment, the diameter of the pulley, the chosen angular deflection and the initial shortening of the muscles are shown.

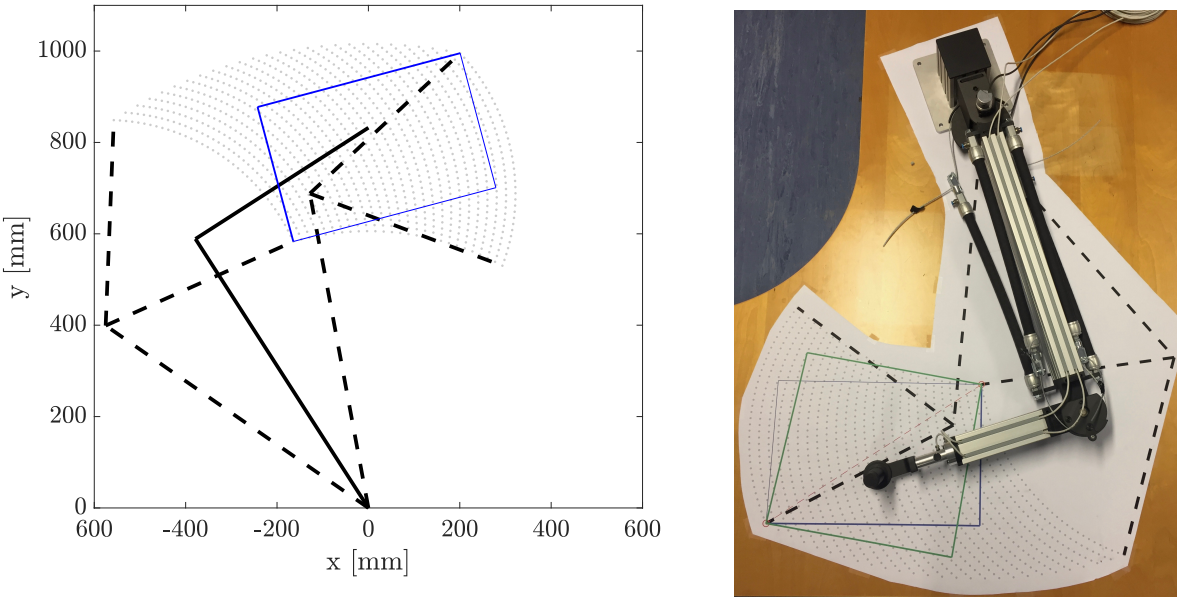


Figure 2.17: Top view of the robot in its workspace.

Parameters	**Joint 1**	**Joint 2**	**Unit**
Length of segments, L	700	450	mm
Pulley diameter, R	140	90	mm
Angular deflection, α	45	65	°
Initial shortening length, δl	55	51	mm

Table 2.2: Key data for the dimensioning of the robot arm.

The initial length of the muscles, l_{init}, depends on the length of the muscles without load l_0. In our case, this length is 400 m. It also depends on an initial shortening length, δl, which allows the cable between the two muscles of a joint to be stretched in the starting situation. In order to know the length of a muscle at the starting point, half of this δl has to be subtracted from the unloaded length. This configuration is shown in FIG. 2.18.

$$\begin{cases} \varepsilon_0 & = \frac{l_0 - l_{init}}{l_0} \\ l_{init} & = l_0 - \left(\frac{\delta l}{2}\right) \end{cases} \tag{2.1}$$

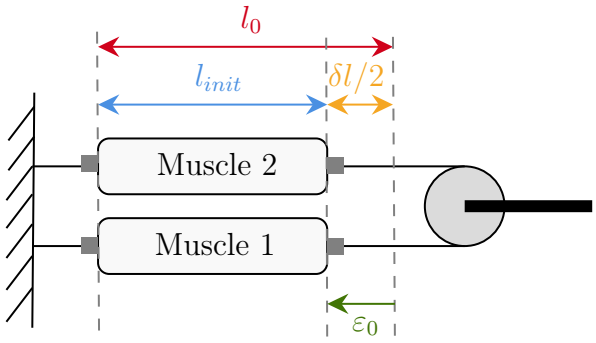


Figure 2.18: Length of muscles in initial position.

The theoretical δl of joints 1 and 2 are 55 mm and 51 mm respectively. However, after fixing the ropes in the pulleys, these theoretical values can be slightly different. In order to obtain taut ropes in the starting situation, the initial pressure to be integrated in each muscle is 2.3 bar. The measured δl value varies slightly depending on the previous use of the arm. Nevertheless, experiments have shown that the level of imperfections observed, due to this difference in values, and therefore the impact is sufficiently low. As a result, the theoretical values are retained.

The δl must be the same on both sides of the joint. Any shortening on one side must induce a symmetrical lengthening on the other, otherwise the rope will slacken. If this happens, there is a risk of losing contact with the pulley and the rope may become misplaced.

When the muscles contract and the pulley rotates through an angle α, the contraction ratios of each muscle are given by the following expressions:

$$\begin{cases} \varepsilon_1 &= \varepsilon_0 - \frac{R\ \alpha}{l_0} \\ \varepsilon_2 &= \varepsilon_0 + \frac{R\ \alpha}{l_0} \end{cases} \tag{2.2}$$

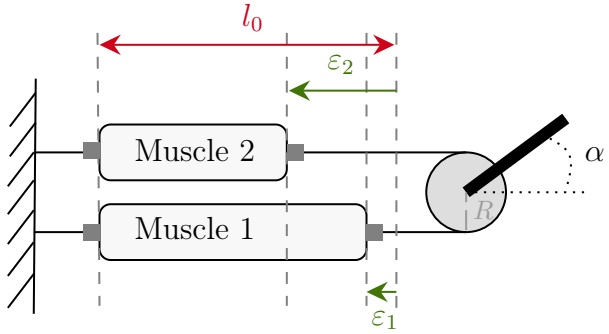


Figure 2.19: Elongation of the muscles during movement.

In the preliminary experimental phase, the robot was mechanically calibrated so that the sensor reading was correct. In the initial position, with identical pressure in each muscle, the arm forms a 90° angle at the elbow. When starting the software, the handle must be on its starting point. Thus, the value of the angular sensors will correspond to 0° initially.

2.4 Summary

The robotic arm is based on the SCARA model with 2 DoFs. It consists of two joints, each activated by two pneumatic artificial muscles. The system contains two angular sensors and a force sensor. The sensor measurements are communicated to the LabView software via a CompactRIO. This also regulates the pressure values to be sent to each muscle. The robot frame is located in the Cartesian reference space and the initial position of the robot is formed by a 90° angle at the elbow.

As previously stated, the artificial muscle contracts when subjected to a certain pressure and load. A relationship between these parameters that models the behaviour of the muscle must be found. The following chapter deals with this task.

Chapter 3

Modelling of an artificial muscle

In order to drive the robot, we need to know what data to provide as input to the system. Therefore we need to identify the inverse model of the robot. This chapter first describes the direct model of the robot, and the method used to invert it. Then, the procedure to find the model of an artificial muscle is described. This can be either based on physics and theory or experimentally. In our case, the methodology was applied to the Festo muscle. Once a suitable model has been identified, the final inverse model is found.

3.1 Direct & Inverse Model

The model seeks to regulate the two parameters: the position (x, y) or the force components (F_x, F_y) at the interface, and the two stiffness components (K_x, K_y). Force and stiffness are not equivalent physical variables. Force is a physical variable that is felt anyway.

Stiffness

Stiffness is a physical variable felt if there is a movement, or if a perturbation is applied to the device. In other words, the stiffness is the feeling at the handle when the robot is moved from an equilibrium position. The overall stiffness felt at the handle is created via the joint stiffnesses of the two muscle pairs. For this type of robot, the global stiffness has an elliptical shape, respresented in Fig. 3.1 [9].

The joint stiffness corresponds to the ratio between the opposing torque and the angular displacement and is therefore measured in $\mathrm{N\,m\,rad^{-1}}$.

$$K_{joint} = \frac{-\tau_{joint}}{\alpha_{joint}} \tag{3.1}$$

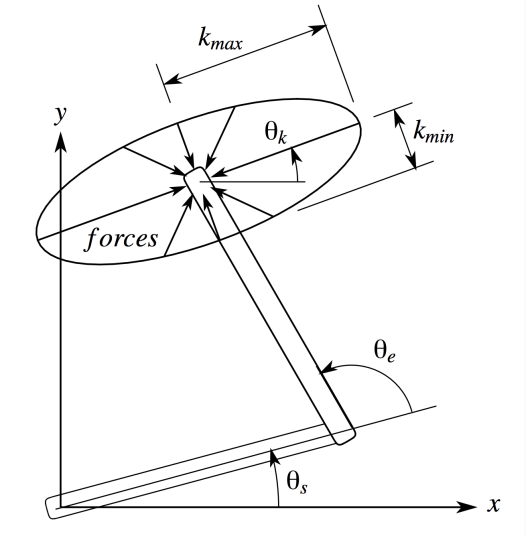


Figure 3.1: Stiffness ellipse representation of stiffness.

A control in stiffness allows to master this robot rigidity.

3.1.1 Direct model

In the direct model, the inputs to the system are the pressures sent to each muscle. For each joint, this pressure couple provides an angular position (α_1, α_2), a joint torque (τ_1, τ_2) and a joint stiffness (K_1, K_2). The combination of each joint finally gives the output of the system as the Cartesian position of the handle (x, y), a force (F_x, F_y) and a certain stiffness (K_x, K_y) felt at it. The transition from the joint space to the Cartesian space of the handle is done via the Jacobian operator. Its expression is given in Appendix B, as well as the development to obtain it. The schematic of the direct model of the robot is shown in Fig. 3.2. It is composed of two boxes with 1 DoF.

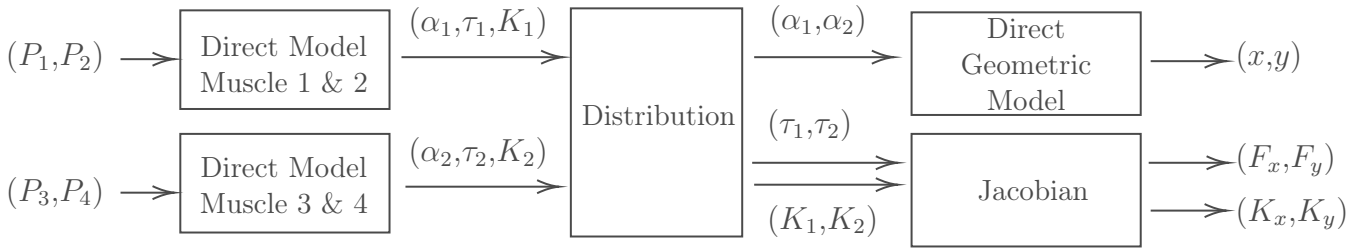


Figure 3.2: Schematic of the direct model of the robot.

Direct model

Inputs	Intermediate Outputs	Outputs
P_i, Pressures sent to each muscle	α_j, Joint angular positions	x, y, Cartesian position of the handle
	τ_j, Joint torques	$F_{x,y}$, Force at the handle
	K_j, Joint stiffnesses	$K_{x,y}$, Stiffness at the handle

Table 3.1: Inputs and outputs of the direct model.

3.1.2 Inverse model

To control the robot in position or in force, and to be able to add a stiffness control law, the model must be inverted. It is necessary to determine the pressures to be sent to each muscle in order to feel a certain force and stiffness at the handle. The schematic of the inverse model of the robot is shown in Fig. 3.3. For a force control with an impedance control law, the inputs to the system are the desired force and stiffness at the handle in the x and y directions. The angular data (α_1, α_2) are provided by the data collected by the angular sensors at each joint. This model allows to know the pressures to be applied in each muscle according to the desired force and stiffness of the handle.

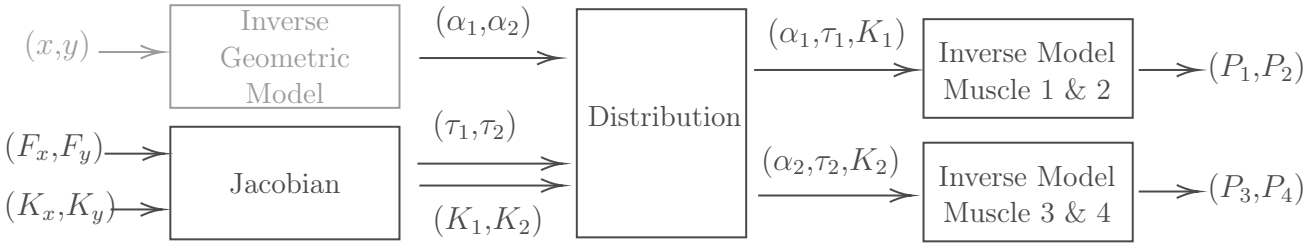


Figure 3.3: Schematic of the inverse model of the robot.

Inverse model

Inputs	Intermediate Outputs	Outputs
x, y, Cartesian position of the handle	α_j, Joint angular positions	P_i, Pressures sent to each muscle
OR $F_{x,y}$, Force at the handle	τ_j, Joint torques	
$K_{x,y}$, Stiffness at the handle	K_j, Joint stiffnesses	

Table 3.2: Inputs and outputs of the inverse model.

3.1.3 Decoupling of the model for each joint

In order to validate this scheme, one must be convinced that it is acceptable to treat each joint independently. It is then sufficient to use a simple Jacobian operator to move from joint space to Cartesian space and vice versa. The Jacobian of the SCARA manipulator with 2 DoFs is of the following form[1], where the θ angles are defined in the robot frame of reference (Sec. 2.2):

$$J = \begin{pmatrix} -L_1 \sin(\theta_1) - L_2 \sin(\theta_1 + \theta_2) & -L_2 \sin(\theta_1 + \theta_2) \\ L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2) & L_2 \cos(\theta_1 + \theta_2) \end{pmatrix} \tag{3.2}$$

This operator is used to establish the relationship between the speed of the end-effector, i.e. the robot's handle, and those of the joints.

$$\begin{pmatrix} \dot{x_{ee}} \\ \dot{y_{ee}} \end{pmatrix} = J \begin{pmatrix} \dot{\theta_1} \\ \dot{\theta_2} \end{pmatrix} \tag{3.3}$$

By the principle of virtual work, if δX_{ee} and $\delta\theta$ represent infinitesimal virtual displacements in task space and joint space respectively, the following relationship is given.

$$\delta X_{ee} = J \ \delta\theta \tag{3.4}$$

If the virtual work δw of the system is equal to zero, the manipulator is at equilibrum.

$$\begin{aligned} \delta w &= \text{Energy injected} - \text{Energy dissipated} \qquad (3.5) \\ &= F^T \ \delta X_{ee} - \tau^T \ \delta\theta \\ &= (F^T J - \tau^T) \ \delta\theta = 0 \\ \tau &= J^T F \qquad (3.6) \end{aligned}$$

[1] As a reminder, the mathematical development of the Jacobian operator is detailed in Appendix B

So, the relationship between an external force applied to the handle and the torque produced in the joints to keep the robot balanced is also established with the Jacobian operator. In the case of the robot used in this project, i.e. a SCARA with 2 DoFs, there are two joints, and thus two torques, which induce a force at the handle in a horizontal plane.

$$\begin{pmatrix}\tau_1\\ \tau_2\end{pmatrix} = J^T \begin{pmatrix}F_x\\ F_y\end{pmatrix} \tag{3.7}$$

$$\Leftrightarrow \begin{pmatrix}F_x\\ F_y\end{pmatrix} = \left(J^T\right)^{-1} \begin{pmatrix}\tau_1\\ \tau_2\end{pmatrix} \tag{3.8}$$

To find a relationship between the stiffness at each joint and the stiffness at the handle, the property of joint compliance is used. Indeed, if they are compliant, the end-effector follows Hooke's law.

$$\begin{pmatrix}F_x\\ F_y\end{pmatrix} = K_{ee}\ \delta X_{ee} = K_{ee} J\ \delta\theta \tag{3.9}$$

$$\Leftrightarrow \begin{pmatrix}\tau_1\\ \tau_2\end{pmatrix} = J^T K_{ee} J\ \delta\theta = K_j\ \delta\theta \tag{3.10}$$

K_j represents the stiffness matrix of the joints. Each joint must have no common elements in order to be able to treat them completely independently. The matrix K_j is therefore a diagonal matrix. K_{ee} is the stiffness matrix of the end-effector. It has elements according to x, y and a combination of both.

$$K_j = \begin{pmatrix}K_1 & 0\\ 0 & K_2\end{pmatrix} = J^T K_{ee} J \tag{3.11}$$

$$K_{ee} = \begin{pmatrix}K_x & K_{xy}\\ K_{yx} & K_y\end{pmatrix} = J^{-T} K_j J^{-1} \tag{3.12}$$

The expression of K_j (Eq. (3.11)) can be developed.

$$\begin{aligned}
K_j &= J^T K_{ee} J \\
&= \begin{pmatrix}J_{11} & J_{21}\\ J_{12} & J_{22}\end{pmatrix} \begin{pmatrix}K_x & K_{xy}\\ K_{yx} & K_y\end{pmatrix} \begin{pmatrix}J_{11} & J_{12}\\ J_{21} & J_{22}\end{pmatrix} \\
&= \begin{pmatrix}J_{11}K_x + J_{21}K_{yx} & J_{11}K_{xy} + J_{21}K_y\\ J_{12}K_x + J_{22}K_{yx} & J_{12}K_{xy} + J_{22}K_y\end{pmatrix} \begin{pmatrix}J_{11} & J_{12}\\ J_{21} & J_{22}\end{pmatrix} \\
&= \begin{pmatrix}J_{11}^2K_x + J_{11}J_{21}(K_{yx}+K_{xy}) + J_{21}^2K_y & J_{11}J_{12}K_x + J_{21}J_{12}K_{yx} + J_{11}J_{22}K_{xy} + J_{21}J_{22}K_y\\ J_{11}J_{12}K_x + J_{21}J_{12}K_{yx} + J_{11}J_{22}K_{xy} + J_{21}J_{22}K_y & J_{12}^2K_x + J_{12}J_{22}(K_{yx}+K_{xy}) + J_{22}^2K_y\end{pmatrix}
\end{aligned}$$

If $K_{yx} = K_{xy}$, which is always true for a conservative system, the expression of K_j can be rewritten.

$$K_j = \begin{pmatrix}J_{11}^2K_x + 2J_{11}J_{21}K_{xy} + J_{21}^2K_y & J_{11}J_{12}K_x + (J_{21}J_{12}+J_{11}J_{22})K_{xy} + J_{21}J_{22}K_y\\ J_{11}J_{12}K_x + (J_{21}J_{12}+J_{11}J_{22})K_{xy} + J_{21}J_{22}K_y & J_{12}^2K_x + 2J_{12}J_{22}K_{xy} + J_{22}^2K_y\end{pmatrix} \tag{3.13}$$

By identifying the terms with the joint stiffness matrix $\begin{pmatrix} K_1 & 0 \\ 0 & K_2 \end{pmatrix}$, an expression for K_1 and K_2, depending on K_x and K_y, is found.

$$K_{xy} = \frac{-J_{11}J_{12}K_x - J_{21}J_{22}K_y}{J_{11}J_{22} + J_{12}J_{21}} \tag{3.14}$$

$$\begin{aligned}
K_1 &= J_{11}^2 K_x + J_{21}^2 K_y - 2J_{11}J_{21}\left(\frac{J_{11}J_{12}K_x + J_{21}J_{22}K_y}{J_{11}J_{22} + J_{12}J_{21}}\right) \\
&= \frac{[J_{11}^2(J_{11}J_{22} + J_{12}J_{21}) - 2J_{11}^2 J_{21}J_{12}]K_x + [J_{21}^2(J_{11}J_{22} + J_{12}J_{21}) - 2J_{11}J_{21}^2 J_{22}]K_y}{J_{11}J_{22} + J_{12}J_{21}} \\
&= \frac{J_{11}^2(J_{11}J_{22} - J_{21}J_{12})K_x - J_{21}^2(J_{11}J_{22} - J_{12}J_{21})K_y}{J_{11}J_{22} + J_{12}J_{21}} \\
&= \frac{(J_{11}J_{22} - J_{12}J_{21})(J_{11}^2 K_x - J_{21}^2 K_y)}{J_{11}J_{22} + J_{12}J_{21}}
\end{aligned} \tag{3.15}$$

$$\begin{aligned}
K_2 &= J_{12}^2 K_x + J_{22}^2 K_y - 2J_{12}J_{22}\left(\frac{J_{11}J_{12}K_x + J_{21}J_{22}K_y}{J_{11}J_{22} + J_{12}J_{21}}\right) \\
&= \frac{[J_{12}^2(J_{11}J_{22} + J_{12}J_{21}) - 2J_{12}^2 J_{11}J_{22}]K_x + [J_{22}^2(J_{11}J_{22} + J_{12}J_{21}) - 2J_{22}J_{12}^2 J_{21}]K_y}{J_{11}J_{22} + J_{12}J_{21}} \\
&= \frac{J_{12}^2(J_{12}J_{21} - J_{11}J_{22})K_x - J_{22}^2(J_{12}J_{21} - J_{11}J_{22})K_y}{J_{11}J_{22} + J_{12}J_{21}} \\
&= \frac{(J_{11}J_{22} - J_{12}J_{21})(J_{22}^2 K_x - J_{12}^2 K_y)}{J_{11}J_{22} + J_{12}J_{21}}
\end{aligned} \tag{3.16}$$

Finally, these equations show that for a desired force (F_x,F_y), and a stiffness (K_x,K_y) at the handle, it is possible to find the torques (τ_1,τ_2) and stiffness (K_1,K_2) at each joint. This reduces the complexity of the problem.

3.1.4 Model inversion method

It is now possible to find the pressures P_i and P_{i+1} to be applied in each pair of muscles in order to induce a torque τ_j and a stiffness K_j at each joint. To do this, we need to find a model of the muscle which shows the relationship between force, pressure and elongation, and expressed as $F = f(\varepsilon, P)$. This step is detailed and done in the next subsection. The model must then be inverted while introducing the notions of torque and stiffness. The following method is used. As a reminder, the contraction ratios equations for a pair of muscles of a joint are expressed as follows:

$$\begin{cases} \varepsilon_1 &= \varepsilon_0 - \frac{R\ \alpha}{l_0} \\ \varepsilon_2 &= \varepsilon_0 + \frac{R\ \alpha}{l_0} \end{cases} \tag{3.17}$$

We start with the torque and joint stiffness equations.

Torque and stiffness equations

$$\tau = R\ (F_2(\varepsilon_2, P_2) - F_1(\varepsilon_1, P_1)) \tag{3.18}$$

$$K = -\frac{\partial \tau}{\partial \alpha} = -R\left(\frac{\partial F_2(\varepsilon_2, P_2)}{\partial \alpha} - \frac{\partial F_1(\varepsilon_1, P_1)}{\partial \alpha}\right) \tag{3.19}$$

If we expand the expression of K:

$$\begin{aligned} K &= -R\left(\frac{\partial F_2}{\partial \varepsilon_2}\frac{\partial \varepsilon_2}{\partial \alpha} - \frac{\partial F_1}{\partial \varepsilon_1}\frac{\partial \varepsilon_1}{\partial \alpha}\right) \\ &= -R\left(\frac{\partial F_2}{\partial \varepsilon_1}\left(\frac{R}{l_0}\right) - \frac{\partial F_1}{\partial \varepsilon_2}\left(-\frac{R}{l_0}\right)\right) \\ &= -\frac{R^2}{l_0}\left(\frac{\partial F_2}{\partial \varepsilon_2} + \frac{\partial F_1}{\partial \varepsilon_1}\right) \end{aligned} \tag{3.20}$$

The model gives the relationship between the force, the deformation and the pressure of the muscle:

$$F = f(\varepsilon, P) \tag{3.21}$$

$$\frac{\partial F}{\partial \varepsilon} = \frac{\partial f}{\partial \varepsilon} = f'(\varepsilon, P) \tag{3.22}$$

Thus, the torque (Eq. (3.18)) and the stiffness (Eq. (3.20)) become:

$$\tau = R\ (f(\varepsilon_2, P_2) - f(\varepsilon_1, P_1)) \tag{3.23}$$

$$K = -\frac{R^2}{l_0}(f'(\varepsilon_2, P_2) - f'(\varepsilon_1, P_1)) \tag{3.24}$$

Inverting the model means inverting f and f'.

$$P = f^{-1}(\varepsilon, F) \Rightarrow P = f^{-1}(\varepsilon, f(\varepsilon, P)) \tag{3.25}$$

$$P = f'^{-1}(\varepsilon, \frac{\partial F}{\partial \varepsilon}) \Rightarrow P = f'^{-1}(\varepsilon, f'(\varepsilon, P)) \tag{3.26}$$

In this case, we can find P_1 with the equation of the stiffness:

$$P_1 = f'^{-1}\left(\varepsilon_1, f'(\varepsilon_2, P_2) + \frac{K\ l_0}{R^2}\right) \tag{3.27}$$

This can be injected in the equation of the torque:

$$\tau = R\left(f(\varepsilon_2, P_2) - f\left(\varepsilon_1, f'^{-1}\left(\varepsilon_1, f'(\varepsilon_2, P_2) + \frac{K\ l_0}{R^2}\right)\right)\right) \tag{3.28}$$

Then, P_2 can be isolated in the last equation. Finally, it is possible to find P_1.

3.2 Model of an artificial muscle

When an artificial pneumatic muscle is subjected to pressure, it shortens by a certain length. This depends on the force to which the muscle is subjected. To relate the pressure introduced into the muscle, the force applied to it and the elongation, a model must be found. Different approaches are considered. The model can be found either theoretically based on the physics of the muscle or experimentally. Festo's muscles have a very specific physics and behaviour. It is therefore necessary to determine the most optimal approach for our particular case.

3.2.1 Physical model

The first model is based on the real physics of the muscles. To create a physical model, it is necessary to know exactly the internal structure of the muscle. However, such information is not always available when the muscle is not made by the user. The artificial pneumatic muscles of reference are the McKibben muscles. Based on the theory of thin-walled rubber tubes, the physical model of these muscles can be obtained. The static model of the muscle was stated earlier.

Physical model

$$F(\varepsilon, P) = (\pi r_0^2)\ P\ \left[a(1-k\varepsilon)^2 - b\right] \qquad \text{with} \qquad \begin{cases} \varepsilon &= \frac{(l_0-l)}{l_0} \\ a &= \frac{3}{\tan^2(\alpha_0)} \\ b &= \frac{1}{\sin^2(\alpha_0)} \end{cases}$$

where F is the axial contraction force induced by the pressure P and causing an elongation ε, l_0 is the initial length of the muscle, α_0 is the initial muscle braid angle, r_0 is the initial muscle radius. In this expression, k is an experimentally derived empirical factor. It allows to take into account two irregular effects of the muscle: a bound effect related to the fact that the muscle does not keep its cylindrical shape near the tip when it contracts, and the non-integral transmission of the pressure force in the muscle weave [31].

In the initial physical model, k is a parameter dependent on P which takes the form:

$$k = a_k e^{-P} + b_k \tag{3.29}$$

However, this expression makes the model difficult to invert because of the P-dependent exponential. A new definition of the parameter k is then proposed, with the aim of obtaining a polynomial equation that will make the model more easily invertible. The new relation is

of the following form[2]:

$$k = \frac{c}{P - d} \tag{3.30}$$

The empirical parameters c and d are two dimensionless real numbers.

In order to simplify the model further, the term $(k\varepsilon)^2$ can be neglected in front of the term $(2k\varepsilon)$. The model thus becomes:

$$F(\varepsilon, P) = (\pi r_0^2)\ P\ [(a-b) - 2ak\varepsilon] \tag{3.31}$$

The force equations and their derivatives for the muscles of a joint are as follows:

$$\begin{cases} F_1(\varepsilon_1, P_1) & = (\pi r_0^2)\ P_1 \left[(a-b) - 2a\frac{c}{P_1 - d}\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)\right] \\ F_2(\varepsilon_2, P_2) & = (\pi r_0^2)\ P_2 \left[(a-b) - 2a\frac{c}{P_2 - d}\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)\right] \end{cases} \tag{3.32}$$

$$\begin{cases} \frac{\partial F_1(\varepsilon_1, P_1)}{\partial \alpha} & = 2a(\pi r_0^2)\frac{R}{l_0}\frac{cP_1}{P_1 - d} \\ \frac{\partial F_2(\varepsilon_2, P_2)}{\partial \alpha} & = -2a(\pi r_0^2)\frac{R}{l_0}\frac{cP_2}{P_2 - d} \end{cases} \tag{3.33}$$

By introducing the derivative expressions into the stiffness equation (Eq. (3.19)), a relationship linking the two pressures is found.

$$\begin{aligned} K &= -R\ \left(\frac{\partial F_2(\varepsilon_2, P_2)}{\partial \alpha} - \frac{\partial F_1(\varepsilon_1, P_1)}{\partial \alpha}\right) \\ &= -\underbrace{\frac{2ac(\pi r_0^2)R^2}{l_0}}_{Q}\left(\frac{-P_2}{P_2 - d} - \frac{-P_1}{P_1 - d}\right) \\ &= Q\frac{2P_1P_2 - d(P_1 + P_2)}{(P_1 - d)(P_2 - d)} \end{aligned} \tag{3.34}$$

$$\Rightarrow P_1 = \frac{(K - Q)\cdot d \cdot P_2 - K \cdot d^2}{(K - 2Q)\cdot P_2 - d\cdot(K - Q)} \tag{3.35}$$

By inserting the expression for P_1 (Eq. (3.35)) into the torque equation (Eq. (3.18)), an equation depending only on P_2 is obtained.

$$\tau = R\ (F_2(\varepsilon_2, P_2) - F_1(\varepsilon_1, P_1)) \tag{3.36}$$

$$\begin{aligned} \frac{\tau}{(\pi r_0^2)\cdot R} &= (a-b)(P_2 - P_1) + 2ac\frac{P_1}{P_1 - d}\varepsilon_1 - 2ac\frac{P_2}{P_2 - d}\varepsilon_2 \\ \frac{\tau}{(\pi r_0^2)\cdot R} &= (a-b)\left(P_2 - \left(\frac{(K-Q)\cdot d\cdot P_2 - K\cdot d^2}{(K-2Q)\cdot P_2 - d\cdot(K-Q)}\right)\right) \\ &\quad + 2ac\frac{\left(\frac{(K-Q)\cdot d\cdot P_2 - K\cdot d^2}{(K-2Q)\cdot P_2 - d\cdot(K-Q)}\right)}{\left(\frac{(K-Q)\cdot d\cdot P_2 - K\cdot d^2}{(K-2Q)\cdot P_2 - d\cdot(K-Q)}\right) - d}\varepsilon_1 - 2ac\frac{P_2}{P_2 - d}\varepsilon_2 \end{aligned} \tag{3.37}$$

[2] *Note:* Another relationship can be chosen, but this does not change the following procedure.

The result is a 4^{th}-order polynomial equation in P_2 which must be solved analytically to find the value of P_2, and finally that of P_1.

However, the Festo muscle is physically different from the McKibben muscle. The braid of the Festo muscle is embedded in the rubber, while that of the McKibben muscle is external. The model presented here is valid on the assumption that this difference does not alter the physical properties of the two muscles and that they behave in the same way, which is not necessarily true.

3.2.2 Experimental Model

Another way to create the muscle model is to do it experimentally. The muscle must be subjected to different static situations, that is, to different loads and different pressures. Then, the data must be recorded one by one to finally create a mathematical model by interpolating the data. This methodology was followed by Festo to provide an experimental model describing the behaviour of the muscle. It is in the form of a polynomial of order 5 containing 6 parameters. The value of the coefficients are listed in Tab. 3.3[3].

Festo model

$$F(\varepsilon, P) = (P - a)\ b + c\ l_0^3\ \varepsilon_0^3 + d\ l_0^2\ \varepsilon_0^2 + (e + f\ P)\ l_0\ \varepsilon_0 \tag{3.38}$$

a	b	c	d	e	f
1e5	2.612e-3	-1.452e6	2.775e5	-1.381e4	-2.048e-2

Table 3.3: Values of the coefficients of the experimental model provided by Festo.

The Festo muscle datasheet provides a graph of this model showing the relationship between force (in [N]) and contraction (in [%] of the original length of the muscle) according to the pressure applied to the muscle.

[3]The variables used are specific to this model, and should not be confused with those present in the rest of the report.

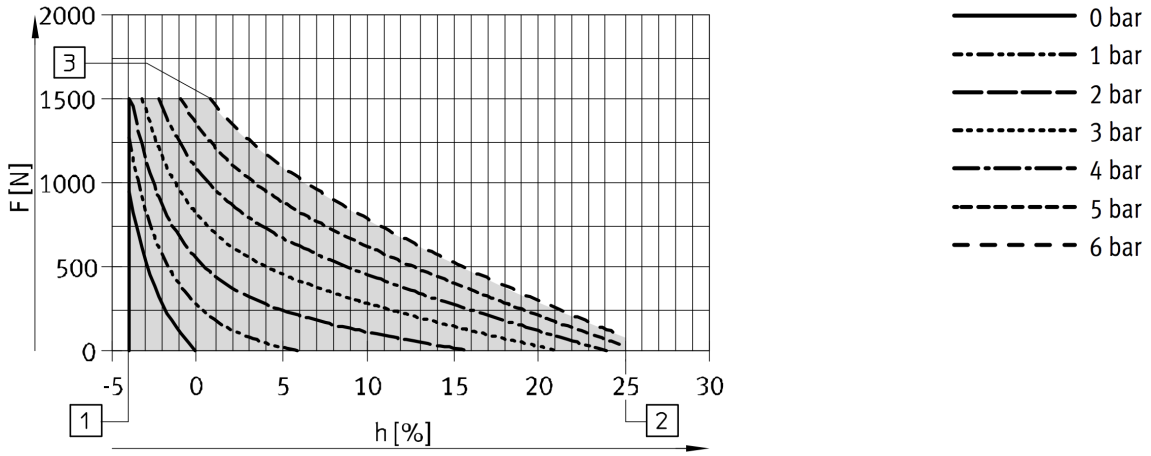
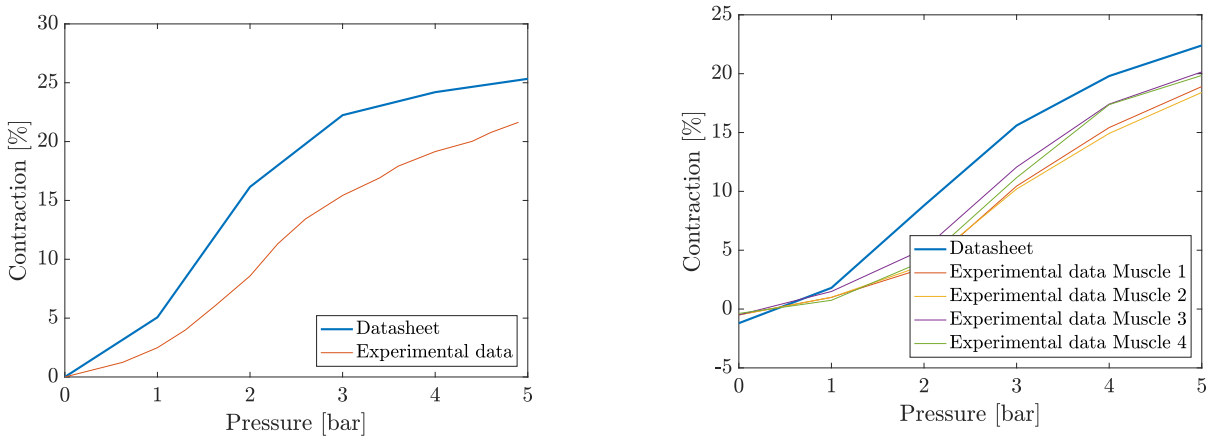


Figure 3.4: Permissible force F [N] as a function of the contraction h [%] in the nominal length, provided by the Festo Fluidic muscle DMSP datasheet [10].

However, some preliminary experimental tests show that this graph does not give the data for the particular muscles used. FIG. 3.5 illustrates the relationship between the pressure sent to a muscle and its elongation when subjected to a fixed load (0 N and 100 N respectively). They compare the data provided by the model of Festo and the experimental data made on one or more muscles of our robot. A significant difference is observed, especially at low force. A note in the datasheet of these muscles states that the actual value of the force as a function of the contraction can vary according to the product characteristics and the ambient conditions. It is therefore not surprising to observe differences between the model of Festo and our experimental data.



(a) 0 N

(b) 100 N

Figure 3.5: Comparison of the theoretical model and experimental data for a fixed load.

Given the significant and not negligible difference between the empirical model provided by Festo and the experimental data, it was decided to create a new experimental model. In order to achieve this task, the following procedure is used: A muscle is hung in a vertical position and is subjected to a load. The load starts at 0 N and increases by 20 N with each experiment until a value of 1000 N. For each load, a pressure of 1 to 5 bar is applied to the muscle. The shortening of the muscle is measured with a calliper, accurate to a hundredth of a millimetre for each test. FIG. 3.6 shows the experimental set-up.

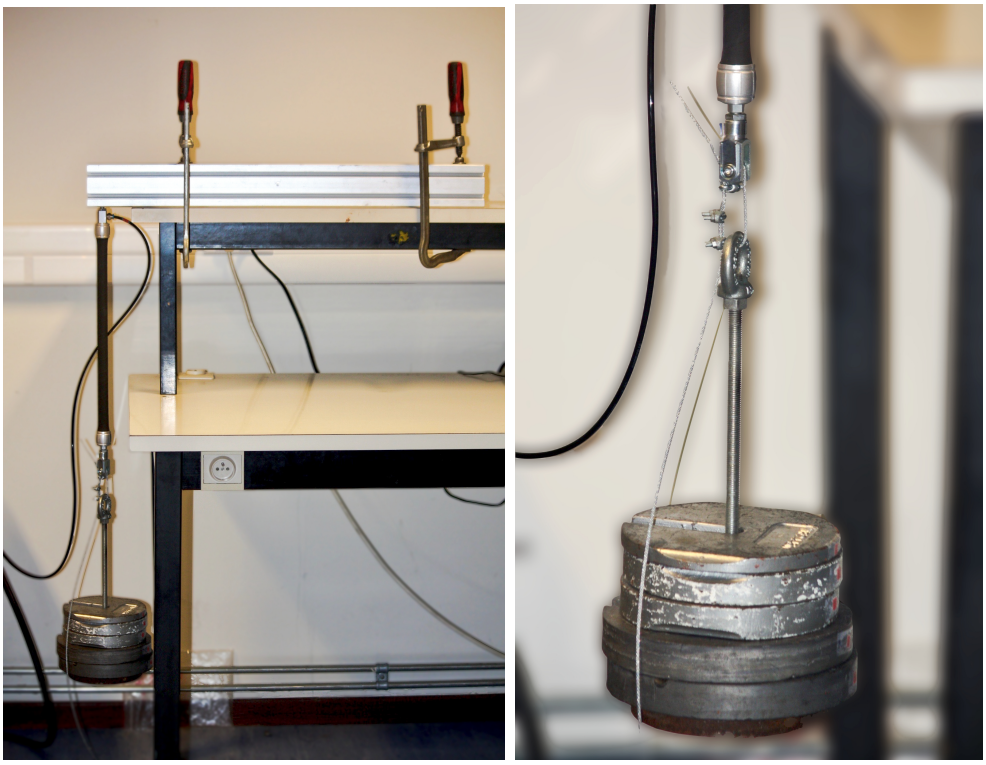


Figure 3.6: Experimental set-up.

All experimental data collected are entered into a MatLab program. The *polyfitn* function allows a polynomial regression to be performed which fits all the data. As a result, a new empirical model of the force as a function of the contraction and pressure of the muscles is found. In our case, the muscle subjected to the experiment is the Festo muscle used in the project. The results obtained for it are shown in FIG. 3.7.

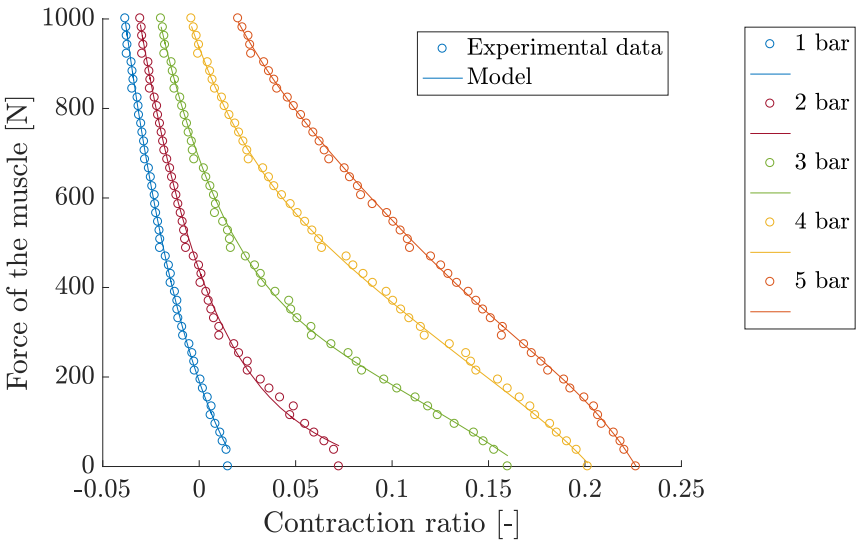


Figure 3.7: Force F [N] as a function of the contraction ε of the new model.

The model is a 5$^{\text{th}}$-order polynomial in order to predict the change in curvature at low force. The order of P has been kept to 1$^{\text{st}}$-order to facilitate the inversion of the model and to keep an approach equivalent to the model provided by Festo. Forces are expressed in Newton [N], pressures in Pascal [Pa] and elongations in relative values ($\varepsilon = \frac{l_0 - l}{l_0}$). The values of the coefficients are shown in Tab. 3.4[4].

Experimental model

$$F(P,\varepsilon) = aP\varepsilon^4 + bP\varepsilon^3 + cP\varepsilon^2 + dP\varepsilon + eP + f\varepsilon^5 + g\varepsilon^4 + h\varepsilon^3 + i\varepsilon^2 + j\varepsilon + k \quad (3.39)$$

a	b	c	d	e
-1.235	0.8748	-0.1696	0.003	0.0025

f	g	h	i	j	k
-6394597.5996	4353194.3094	-1296730.9029	188507.2445	-12713.2922	-58.1436

Table 3.4: Values of the polynomial coefficients.

The graph of the model shows that the greater the shortening, the lower the force capacity. It is interesting to note, by observing the curves, that a massive force is developed for negative elongations. It is a force that cannot be generated, it is done passively. This

[4]Again, the variables used are specific to this model.

force is used as a damper, a physical safety element, similar to a mechanical stop. At constant pressure, there is a strong variation in compliance, which decreases sharply when the system reaches the stop.

At this stage, it is wise to check that the model found does not overfit the experimental data. In other words, it is necessary to check that the model does not simply pass through all the points, but that it gives a valid curve for any other pressure that is not part of the experimental data. To do this, curves for intermediate pressures are drawn. The graph in FIG. 3.8 shows that these new curves (small grey stars) correctly follow the model curves. This proves that the model does not simply overfit the experimental data.

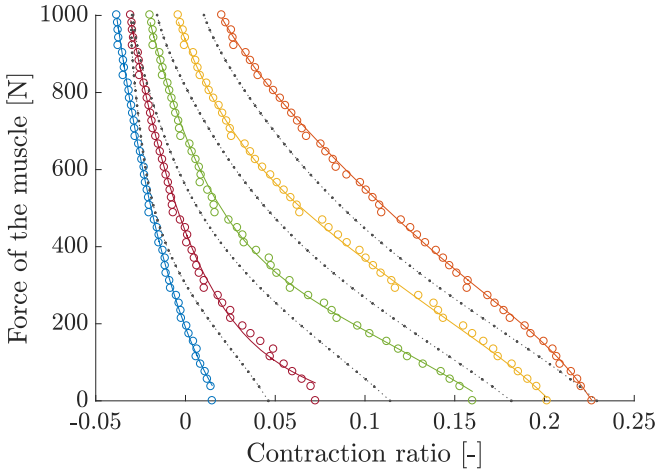


Figure 3.8: Proof that the model is not overfitting the experimental data.

FIG. 3.9 compares the experimental data to the result of the physical model. A slight difference is observed.

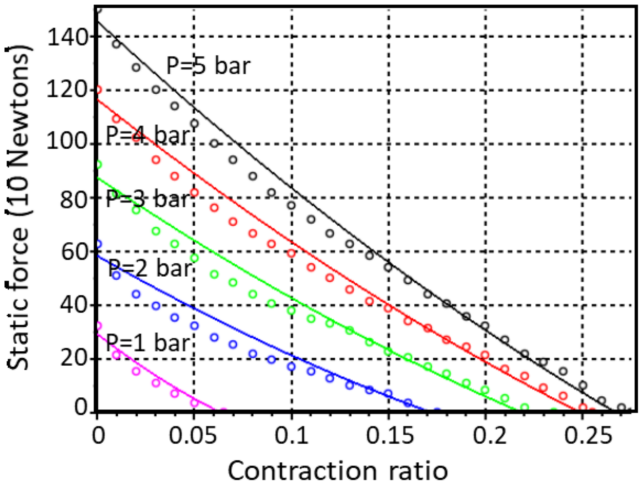


Figure 3.9: Comparison of the physical model (in full line) and the experimental data [30].

One explanation for this phenomenon could be that hysteresis was not taken into account when the data was collected. Although the braid of the Festo muscle is integrated into the rubber, internal friction exists. Indeed, the muscle contracts thanks to the opening of its textile braid. There remains therefore friction between the threads inside the braided sheath [30]. A fairly high hysteresis is therefore inevitable. In practice, two different experiments should be carried out: a force recording with decreasing and increasing contraction rates. This would allow the hysteresis phenomenon to be taken into account.

The physical model created for the McKibben muscles is based on the theory of thin-walled rubber tubes. However, this does not apply in our case because the Festo muscles are thicker, and therefore stronger. This makes the theory complex. In the literature, it is observed that the equations of the model become more complicated and the number of parameters increases. A purely empirical approach with parametric identification is then justifiable. This is what is done in article [1]. The parameter P is kept to 1$^{\text{st}}$-order to be able to invert the model. The paper shows that the model would rise to a polynomial of order 30 to keep a certain accuracy. They avoid this with a contraction term to the 2/3 power. But this term creates an infinite stiffness at the origin. In the article, they distinguish between working below and above 2 bar. However, a model valid for any pressure is needed. The current diversity of these models highlights the difficulty of presenting a model combining a physical background with good accuracy [30]. Thus, the experimental model chosen in our case, containing 11 parameters, is consistent.

To summarise, given the very high non-linearities of the Festo muscle, it is assumed that an empirical and purely mathematical approach is superior to a physical approach. Furthermore, the experimental model is easily analytically invertible and can therefore be used for the feed-forward controller.

The inverse model is found by following again the method described above (Sec. 3.1.4). The force equations and their derivatives for the muscles of a joint are as follows:

$$\begin{cases} F_1(\varepsilon_1, P_1) & = aP_1\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^4 + bP_1\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^3 + cP_1\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^2 + dP_1\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right) + eP_1 \\ & \quad + f\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^5 + g\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^4 + h\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^3 + i\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^2 + j\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right) + k \\ F_2(\varepsilon_2, P_2) & = aP_2\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^4 + bP_2\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^3 + cP_2\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^2 + dP_2\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right) + eP_2 \\ & \quad + f\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^5 + g\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^4 + h\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^3 + i\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^2 + j\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right) + k \end{cases}$$

$$\begin{cases} \frac{\partial F_1(\varepsilon_1, P_1)}{\partial \alpha} & = \frac{R}{l_0}\left[(dP_1 + j)\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right) + 2(cP_1 + i)\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right) + 3(bP_1 + h)\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^2\right. \\ & \quad \left. + 4(aP_1 + g)\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^3 + 5f\left(\varepsilon_0 + \frac{R\alpha}{l_0}\right)^4\right] \\ \frac{\partial F_2(\varepsilon_2, P_2)}{\partial \alpha} & = \frac{R}{l_0}\left[(dP_2 + j)\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right) + 2(cP_2 + i)\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right) + 3(bP_2 + h)\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^2\right. \\ & \quad \left. + 4(aP_2 + g)\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^3 + 5f\left(\varepsilon_0 - \frac{R\alpha}{l_0}\right)^4\right] \end{cases}$$

By introducing the derivative expressions into the stiffness equation (Eq. (3.19)), a relationship linking the two pressures is found.

$$K = -R \left(\frac{\partial F_2(\varepsilon_2, P_2)}{\partial \alpha} - \frac{\partial F_1(\varepsilon_1, P_1)}{\partial \alpha} \right) \tag{3.40}$$

$$= P_2 \cdot C_2 + S_2 + P_1 \cdot C_1 + S_1 \tag{3.41}$$

$$\begin{cases} C_1 &= -\frac{R^2}{l_0} \left[d + 2c \left(\varepsilon_0 - \frac{R\alpha}{l_0} \right) + 3b \left(\varepsilon_0 - \frac{R\alpha}{l_0} \right)^2 + 4a \left(\varepsilon_0 - \frac{R\alpha}{l_0} \right)^3 \right] \\ C_2 &= -\frac{R^2}{l_0} \left[d + 2c \left(\varepsilon_0 + \frac{R\alpha}{l_0} \right) + 3b \left(\varepsilon_0 + \frac{R\alpha}{l_0} \right)^2 + 4a \left(\varepsilon_0 + \frac{R\alpha}{l_0} \right)^3 \right] \\ S_1 &= -\frac{R^2}{l_0} \left[j + 2i \left(\varepsilon_0 - \frac{R\alpha}{l_0} \right) + 3h \left(\varepsilon_0 - \frac{R\alpha}{l_0} \right)^2 + 4g \left(\varepsilon_0 - \frac{R\alpha}{l_0} \right)^3 + 5f \left(\varepsilon_0 - \frac{R\alpha}{l_0} \right)^4 \right] \\ S_2 &= -\frac{R^2}{l_0} \left[j + 2i \left(\varepsilon_0 + \frac{R\alpha}{l_0} \right) + 3h \left(\varepsilon_0 + \frac{R\alpha}{l_0} \right)^2 + 4g \left(\varepsilon_0 + \frac{R\alpha}{l_0} \right)^3 + 5f \left(\varepsilon_0 + \frac{R\alpha}{l_0} \right)^4 \right] \end{cases} \tag{3.42}$$

$$\Rightarrow P_1 = \underbrace{\frac{K - S_1 - S_2}{C1}}_{Q_3} - \underbrace{\frac{C_2}{C_1}}_{Q_4} \cdot P_2 \tag{3.43}$$

By inserting the expression for P_1 (Eq. (3.43)) into the torque equation (Eq. (3.18)), an equation depending only on P_2 is obtained.

$$\tau = R \ (F_2(\varepsilon_2, P_2) - F_1(\varepsilon_1, P_1)) \tag{3.44}$$

$$\begin{aligned} \frac{\tau}{R} = &\left(P_2 \underbrace{(a\varepsilon_2^4 + b\varepsilon_2^3 + c\varepsilon_2^2 + d\varepsilon_2 + e)}_{Q_2(\varepsilon_2)} + \underbrace{(f\varepsilon_2^5 + g\varepsilon_2^4 + h\varepsilon_2^3 + i\varepsilon_2^2 + j\varepsilon_2 + k))}_{Q_1(\varepsilon_2)} \right) \\ &- \left(P_1 \underbrace{(a\varepsilon_1^4 + b\varepsilon_1^3 + c\varepsilon_1^2 + d\varepsilon_1 + e)}_{Q_2(\varepsilon_1)} + \underbrace{(f\varepsilon_1^5 + g\varepsilon_1^4 + h\varepsilon_1^3 + i\varepsilon_1^2 + j\varepsilon_1 + k))}_{Q_1(\varepsilon_1)} \right) \\ = &P_2 \ Q_2(\varepsilon_2) + Q_1(\varepsilon_2) - P_1 \ Q_2(\varepsilon_1) - Q_1(\varepsilon_1) \\ = &P_2 \ Q_2(\varepsilon_2) + Q_1(\varepsilon_2) - (Q_3 - Q_4 \ P_2) \ Q_2(\varepsilon_1) - Q_1(\varepsilon_1) \end{aligned}$$

$$\Rightarrow P_2 = \frac{Q_1(\varepsilon_1) - Q_1(\varepsilon_2) + Q_3 \ Q_2(\varepsilon_1) + \tau/r}{Q_2(\varepsilon_2) + Q_4 \ Q_2(\varepsilon_1)} \tag{3.45}$$

This gives a value for P_2. It can finally be injected into the Eq. (3.43) to find the value of P_1.

3.3 Summary

To control the robot, the inputs to the system must be known. The inputs of the direct model are the pressures to be sent to each muscle. The outputs are the position, the force, and the stiffness felt at the handle. However, this model should be inverted to determine the pressures to be sent to the muscles according to the desired position or force and stiffness. To do this, the inverse model is needed.

The model of the artificial muscle can be found theoretically through muscle physics, or experimentally. In our case, the second option is chosen. Indeed, due to the complex behaviour of the Festo muscle, a purely empirical approach is superior to a physical approach. The final model has 11 parameters to ensure sufficient accuracy, and is easily invertible.

It is important to note that, to reduce the complexity of the problem, the system can be split in two: the two joints can be treated separately. Combining the data at the joints (forces and stiffness), via the Jacobian operator, will give the data related to the handle.

Knowing the muscle model, it is now possible to determine the limits of the system. The next chapter characterises the force and stiffness boundaries of the robotic arm.

Chapter 4

Characterisation of the force and stiffness limits of the robot in its working space.

The primary purpose of the robot is to be used in the rehabilitation sessions for patients who have suffered a stroke. The system is to stage a series of games to motivate the patient to perform assisted movements. It is therefore important to understand the physics and kinematics of the system, and to mechanically characterise the robot's capabilities. Such information are useful in order to inform a future game designer.

This chapter will highlight the physical limitations of the robot. It will first evaluate the force that can be achieved at maximum in all directions on the workspace. Next, a graphical illustration of the relationship between the torque delivered to each joint and its stiffness is proposed, in order to better understand the link between these two elements. Then, a research is done to find the maximum and minimum stiffness that can be achieved on the workspace for zero torque at the joints.

4.1 Mapping of the achievable force in the entire workspace

It is interesting for the analysis of the robot to find the robot's strength capabilities. The objective of this section is to find the maximum force deliverable by the robot in all directions of the workspace. This will enable the programmer to guarantee a maximum force that can be reached over the entire area.

To map the attainable force in all directions, a MatLab programme is developed. The procedure is as follows: For each point in the workspace, two torques at each joint are

calculated. These torques are the maximum torques corresponding to a situation where a muscle is completely contracted, i.e. when it is supplied with a pressure of 5 bar. This means that the maximum absolute pressure in one of the two actuators will determine a maximum torque. In other words, at a given angular position, a maximum pressure corresponds to a maximum force that can be generated in a certain direction. Thus, joint 1 has a maximum torque when muscle 1 is contracted at 5 bar, and another maximum torque when muscle 2 is in turn contracted at 5 bar. The same applies to the second joint and muscles 3 and 4. Each of these torques causes a force, F, at the handle. As previously stated, the transition from joint torque to end-effector force is made via the Jacobian operator, J (Sec. 3.1.1). With this, the torques can be projected into the SCARA kinematic model. It is possible to obtain the contribution of the two muscles that generate torque at the end of the arm. The force components at the handle can be calculated.

Relation between joint torques and end-effector force

$$\tau = J^T \cdot F \tag{4.1}$$

$$\begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} = J^T \begin{pmatrix} F_x \\ F_y \end{pmatrix} \tag{4.2}$$

with, as a reminder, the Jacobian operator:

$$J = \begin{pmatrix} -L_1 \cdot \sin(\theta_1) - L_2 \cdot \sin(\theta_1 + \theta_2) & -L_2 \cdot \sin(\theta_1 + \theta_2) \\ L_1 \cdot \cos(\theta_1) + L_2 \cdot \cos(\theta_1 + \theta_2) & L_2 \cdot \cos(\theta_1 + \theta_2) \end{pmatrix} \tag{4.3}$$

For each of the four possible extreme configurations, a maximum force is transmitted to the handle in a certain direction. A combination of the minimum force for each joint gives the force attainable at a particular point in the area in all directions, when maximum torques are delivered to the joints. This value therefore gives the maximum achievable force in all directions of the workspace. Graphically, this value is the radius of the largest circle centred on the handle inscribed in the parallelogram formed by the four forces.

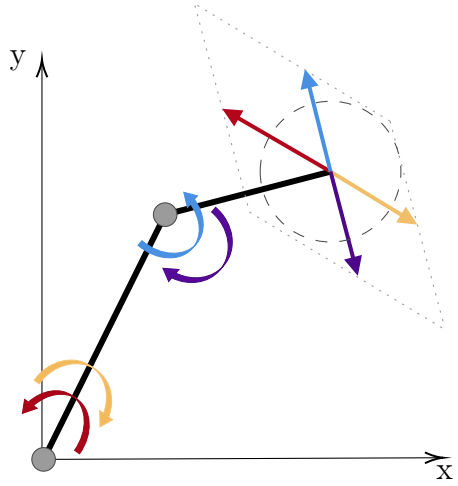


Figure 4.1: Force transmitted to the handle by each torque.

The algorithm of the code used is as follows:

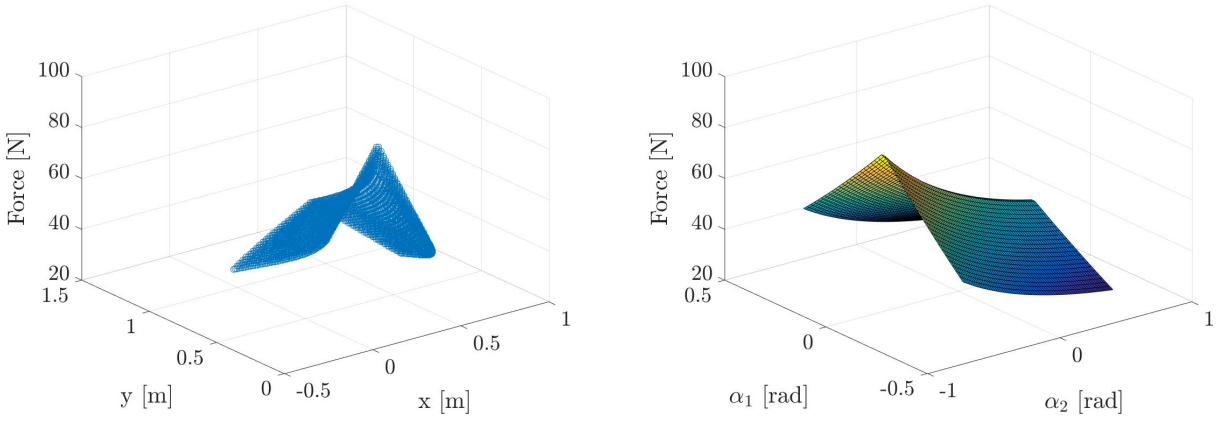
Algorithm 1 Algorithm to find the achievable force in the entire workspace

Result: F_{max} achievable in the entire workspace
Initialization
for $\forall$ $(\alpha_1; \alpha_2)$ **do**
  Find Elongations ε
  Find Jacobian J
  Based on the muscle model, find...

- Case 1: F_{max_1} when Muscle 1 is fully contracted
- Case 2: F_{max_2} when Muscle 2 is fully contracted
- Case 3: F_{max_3} when Muscle 3 is fully contracted
- Case 4: F_{max_4} when Muscle 4 is fully contracted

  Keep F_{max} i.e. the radius of the largest circle centred on the handle inscribed in the parallelogram formed by the 4 forces
end

FIG. 4.2 shows the result of the achievable force in the working area along x and y, and along the angles of the two joints.



(a) According to x and y

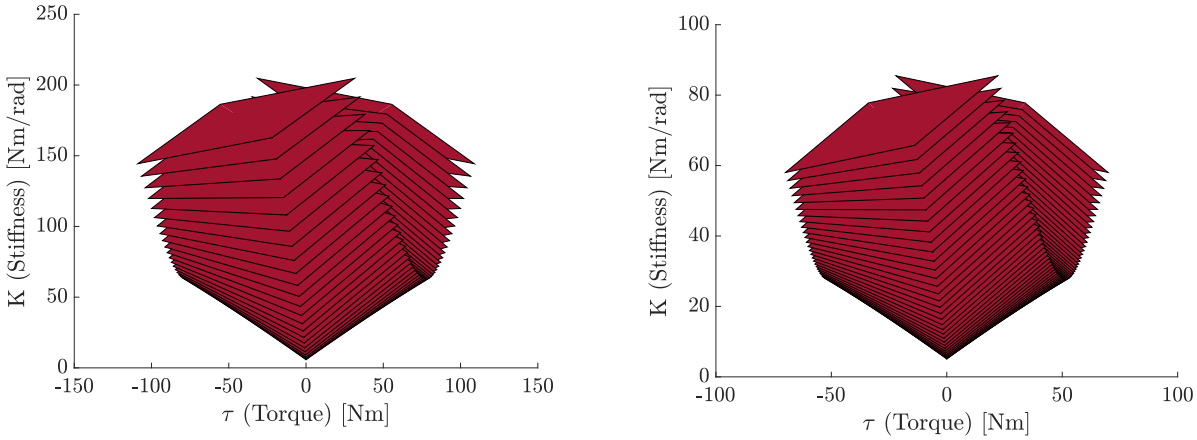
(b) According to α_1 and α_2

Figure 4.2: Maximum force achievable in all directions over the entire working area.

The lowest value in this graph is 26.94 N. It is therefore possible to guarantee a maximum force of about 27 N over the whole area in any direction. A force smaller than this is achievable in the field. To achieve this, it is possible to play on two parameters. Either a reduction in pressure in the agonist muscle which generates the force will create a reduction in force. Or an increase in pressure in the antagonist muscle will decrease the torque but increase the stiffness of the joint.

4.2 Relationship between torque and stiffness

The torque produced at each joint is intrinsically linked to its joint stiffness. The stiffness capacity of the system is significantly dependent on the force capacity of the system. Indeed, both variables depend on P_1 and P_2. The torque is proportional to the term $(P_2 - P_1)$ while the stiffness is proportional to $(P_2 + P_1)$. If a user wants to produce a certain torque, a range of achievable stiffness is bounded. And conversely, if he wants to produce a certain stiffness, this is possible for a defined torque range. The range of attainable values for combinations (Torque τ, Stiffness K) is represented graphically by a diamond. This space changes with robot position and this change is trackable. Fig. 4.3 shows the attainable space as the articular configuration (α angle) of each joint changes. The diamond at the back of the image corresponds to the most negative α, while the diamond in front corresponds to the most open position of the articulation.



(a) Joint 1

(b) Joint 2

Figure 4.3: Reachable range for combinations (τ, K).

The results can be illustrated in the working range of the robot. The joint torques and stiffnesses are thus converted into a force and stiffness in x and y at the end-effector, via the Jacobian operator. The attainable space (diamond) is shown for several selected points (Fig. 4.5): the 4 extremities, the centre, as well as another random point. This shows that this space clearly varies with the position of the robot.

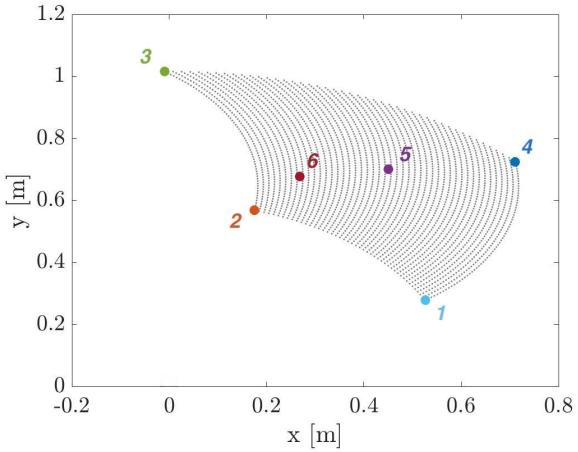
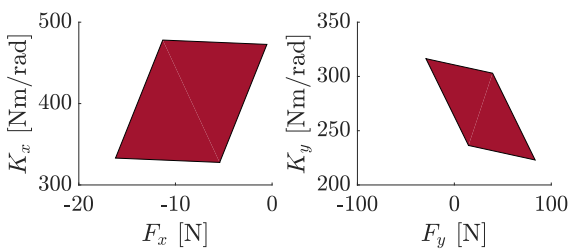
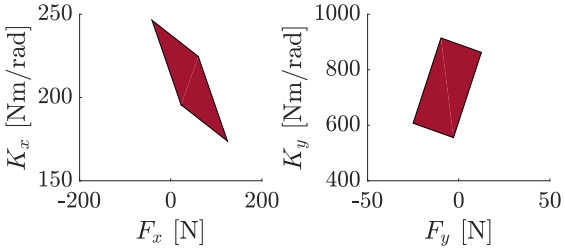


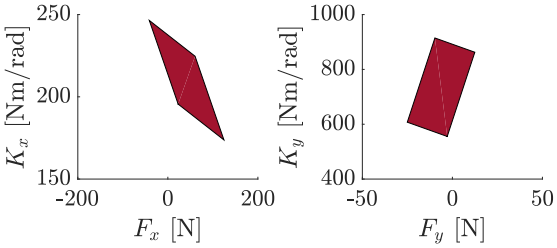
Figure 4.4: Workspace of the robot.



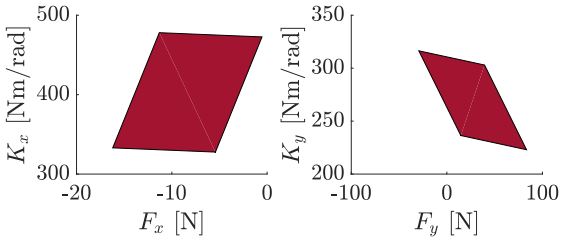
(a) Point 1



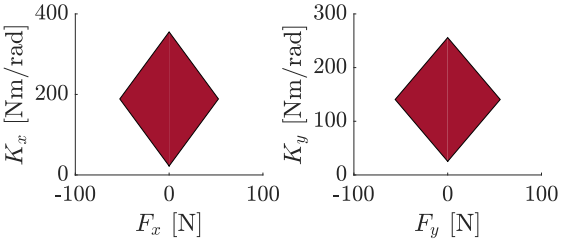
(b) Point 2



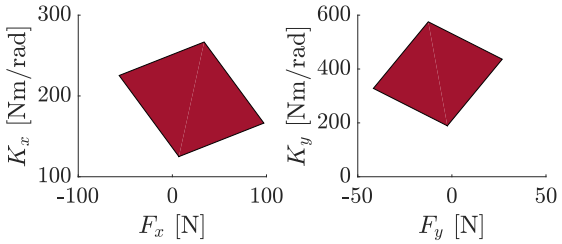
(c) Point 3



(d) Point 4



(e) Point 5



(f) Point 6

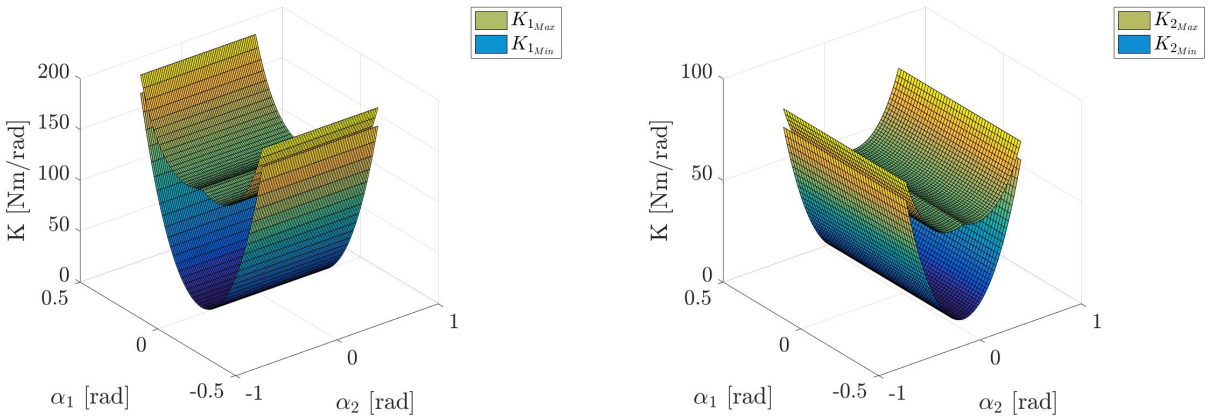
Figure 4.5: Attainable values for combinations ($F_{x,y}$, $K_{x,y}$) for several points of the workspace.

It can be seen that the range of stiffness achievable in the centre is lower than that achievable in the corners. When the robot is at one end, one of the two muscles is already so contracted that the joint is forced to have a high stiffness. Moreover, the centre is the only point in the workspace where low stiffness is possible. Indeed, in this location, the equilibrium position corresponds to a low pressure in all muscles.

The graphs allow to know the physical limits of the system in force for a certain stiffness, and conversely, in stiffness for a certain desired force. An interactive MatLab code allows the user to know, for example, the range of attainable stiffness values for a chosen configuration and force.

4.3 Mapping of the achievable stiffness in the entire workspace.

Returning to the joint domain, it is now possible to look at the maximum and minimum stiffness that can be achieved for each configuration when the robot is transparent, i.e. when the joint torque is zero. The torque-stiffness relationship found above is used to find the range of allowable joint stiffnesses when the joint torque is zero. The results for the two joints are shown in Fig. 4.6.



(a) Joint 1

(b) Joint 2

Figure 4.6: Maximum and minimum achievable stiffness when the robot is transparent.

The values of the minimum stiffness range (top surfaces) and the maximum stiffness range (bottom surfaces) of each joint are listed in Tab. 4.1.

	Joint 1	Joint 2	Unit
Minimum stiffness	[5.91; 180]	[5.151; 73.39]	$\mathrm{N\,m\,rad^{-1}}$
Maximum stiffness	[117.7; 198.1]	[50.31; 82.46]	$\mathrm{N\,m\,rad^{-1}}$

Table 4.1: Range of minimum and maximum stiffness for each joint.

It depends strongly on the position of the robot arm. Therefore, it is not reasonable to set one maximum or minimum achievable value. This would considerably reduce the robot's capabilities. A good strategy would be to vary the stiffness value during therapeutic exercise.

4.4 Summary

The system has limitations in terms of the torque that can be delivered to the joints, and therefore the force that can be provided to the handle. It is important to characterise this limit in order to inform the user about the robot's capabilities. It is therefore necessary to know the maximum force that the robot could deliver in all directions of the workspace. The same work has to be done with the stiffness. However, force and stiffness are two intrinsically linked parameters. It is possible to find the range of values attainable for joint stiffness when a certain torque is required, and vice versa. This interval also varies strongly with the joint configuration.

Having the muscle model and the system limits, all the ingredients are available to implement the robot control. The following chapter presents the low-level control of a joint of the robotic arm, and investigates the performance of the closed-loop system.

Chapter 5

Low-level control of the robot

The development of the robot controller is crucial. The control must allow for smooth and continuous movement, similar to that of humans. This chapter first presents the low-level control used for one joint of the robotic arm. Then, two tasks are performed in order to validate the control of the robot. The first one is an isokinetic task, where the articular configuration is kept constant. In the second task, the robot acts at constant torque, i.e. is controlled to deliver the same torque to the user. During these two tasks, the results obtained when the system is in open loop or closed loop are compared. Finally, the performance of the closed loop system is characterized by looking for its bandwidth through the creation of the Bode diagram.

5.1 Closed loop system control

The system is provided by two actions: a feed-forward action, and a feedback action, performed using a PI controller. The feedback control corrects the errors that are left after the feed-forward commands. Its purpose is to cancel the error in force, based on the measurement. It compares the reference forces in x and y ($F_{x,ref}$, $F_{y,ref}$) with the forces measured by the handle at the interface (F_x, F_y). A regulation is made to minimise this error. The muscle pressures are corrected so that the force measured at the handle matches the desired force. The PI controller is presented in Appendix C. The command $U(s)$ sent to the robot is the combination of both actions and can be expressed in the Laplace domain.

Closed loop system command

$$U(s) = F_{FB} + F_{FF} \tag{5.1}$$

$$= \left(K_p + \frac{K_i}{s}\right)(F_{ref} - F) + F_{FF} \tag{5.2}$$

The complete block diagram of a robot joint control is shown in FIG. 5.1. A desired torque is sent as a reference. It is compared with the measured value using the feedback control. A correction torque increment comes out of the PI controller and is added to the reference value using the feed-forward control. This torque command is inserted into the inverse model which finds the pressures to be delivered to the muscles. The angular sensors and the force sensor return the measured data, which is passed through the Jacobian operator to obtain the joint torque value.

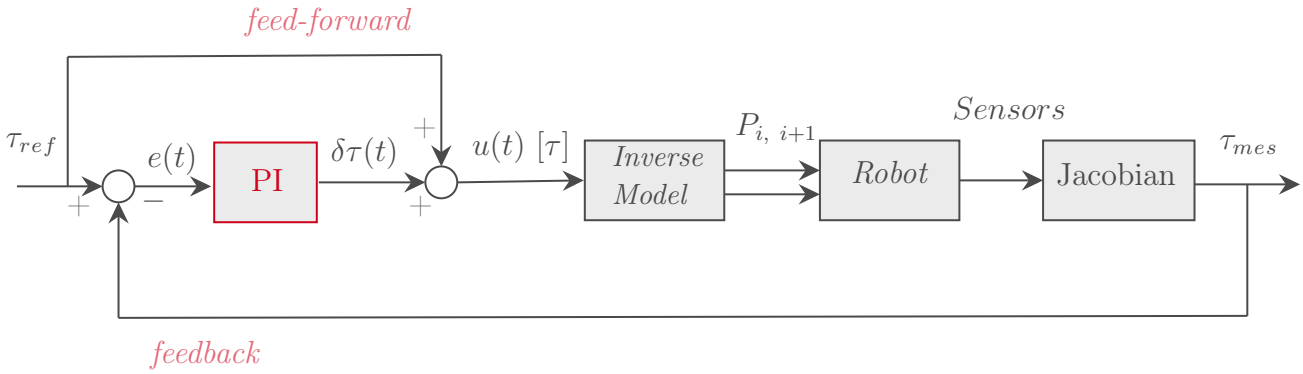


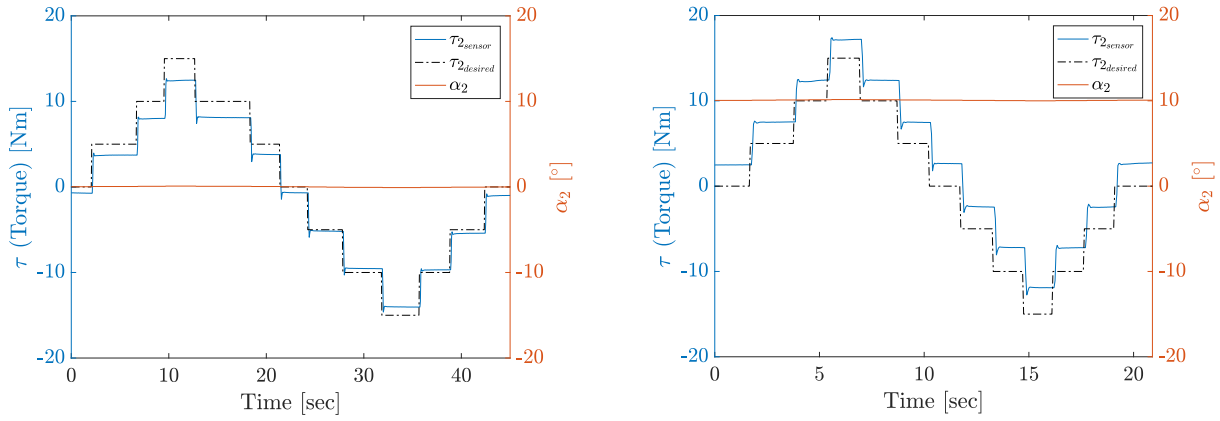
Figure 5.1: Block Diagram of the control of one robot joint.

5.2 Validation task 1: Isokinetic

The inverse experimental model, implemented in the LabView software, enables a first piloting of the robot. The model allows to know the pressures to be sent to each muscle on the basis of desired inputs. In the Cartesian domain, the inputs are the x and y components of the force and stiffness at the handle. In the joint domain, the inputs are the joint torques and stiffness. Several tests can be performed to try to validate the artificial muscle model found. The first test is performed with joint 2. The test is performed on the scale of one axis, and therefore in angular units. Joint 1 is mechanically blocked and does not influence the test. The latter consists of mechanically locking the handle in a certain position, i.e. for a constant α_2, and varying the commanded torque of joint 2, τ_2. The pressures of the two muscles of joint 2 are regulated in order to have and feel an imposed torque. The goal of this test is thus to assess the force control bandwidth with a constant position of the robot handle.

5.2.1 Open loop control

The test is first performed in open loop, i.e. without the presence of the PI controller. The handle is rigidly held for α_2 equal to $[-15°, -10°, -5°, 0°, 5°, 10°, 15°]$ successively. For each test, the torque required varies from $15\,\mathrm{N\,m}$ to $-15\,\mathrm{N\,m}$ in steps of $5\,\mathrm{N\,m}$. Fig. 5.2 illustrates the results of two tests, where α_2 is maintained at a value close to 0° and 10° respectively. This test is used to check how much force is actually felt when a pressure drop is imposed, and whether this force is close to what is mathematically expected.



(a) $\alpha_2 = 0$ (b) $\alpha_2 = 10$

Figure 5.2: Comparison between the desired τ_2 and that measured by the sensor.

We can observe a static error of the torque. Indeed, the value measured by the sensor does not reach the desired values. The other tests, for different α_2 angles, show similar results. The *static error* may be due to different factors:

- The results show a probable inaccuracy of the muscle model.
- There is an inaccuracy due to hidden dynamics that have not been accounted for in the model and that does not depend on the muscle, such as elasticity through the cables or the flexibility of the pieces. The cables themselves are totally inextensible, but they are positioned through three loops (two at the muscle end and one on the pulley). There are effects related to the tensioning of the rope that are strictly not modelled. Those effects can change the actual length of the muscles, so that it is no longer the estimated one. This creates a phase shift between muscle shortening and increasing force. In addition, 3D printed parts have a stiffness that is much lower than the stiffness of a metal part. This can lead to a shortening of the arm, which would again change the actual lengths of the muscles. An improvement of the mechanics in a future version of the prototype will enhance the results.

- The δl's are considered as constant in the model. However, mechanical deformations that are not modelled may cause them to vary in reality. This will modify the values of the pressures to be injected into the muscles.
- There is probably a structural assymmetry of the muscles, due to their manufacture, or of the system itself, due to the method of attachment, the length of the cables, etc.
- The value of τ_2 measured by the sensor may be slightly approximate. Indeed, it comes from the direct model of the muscle, and in particular from the value of F_x and F_y of the force sensor. However, these data have been converted from Volt to Newton via an assumed linear relationship (Sec. 2.1).
- There is a small starting offset which is difficult to remove as it varies with each test.

To improve the results of the open loop system, it might be tempting to add a correction term, i.e. an offset to the command. But experiments have shown that adding an offset will be favourable in some configurations, but unfavourable in others. Therefore, closed loop control appears to be the best strategy to suppress the static error.

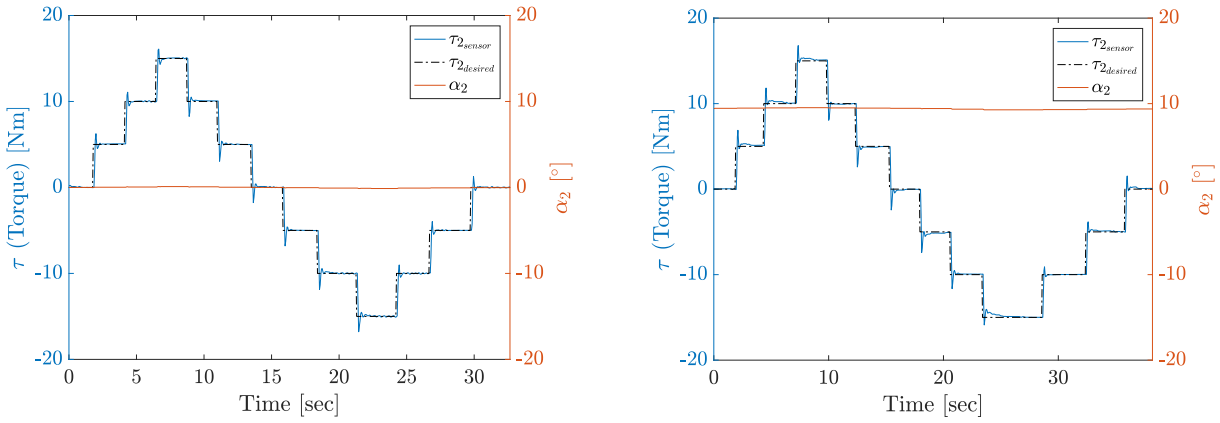
5.2.2 Closed loop control

In order to eliminate, or at least reduce, the errors that exist during open loop control, it is necessary to close the loop. Closed loop control uses the information in force, or rather in torque if we stay in the joint domain. A closed loop control will allow precise movement constraints to be imposed on the patient. For this purpose, a PI controller is implemented. Using the response time and static gain values obtained when the system is in open loop, and using Eqs. (C.15) and (C.16) of the Appendix C, the controller gain values are found mathematically.

$$\begin{cases} K_p & = 0.3 \\ K_i & = 25 \end{cases} \quad \rightarrow \quad \Gamma_i = 0.01 \tag{5.3}$$

The values of these gains can be improved by trial and error. It is interesting to keep in mind some notions about the effect of controller gains. Increasing the proportional gain will decrease the static error and create an overshoot. The effect of the integral gain is to suppress the static error. If there is an error, the integral action integrates it until the output of the integral is constant. As long as the integrand is not zero, the integral increases to remove the error. Increasing the gain K_i induces oscillation around the correct mean value. Increasing the gain K_p reduces these oscillations. Indeed, there is a double integrator in the open loop because a force is controlled to control a position. The proportional action would already have a damping effect and should dampen the oscillations. However, if the proportional gain is too high, the system can become unstable.

The same isokinetic test was performed with the closed loop system. The results are shown in FIG. 5.3.



(a) $\alpha_2 = 0$ (b) $\alpha_2 = 10$

Figure 5.3: Comparison between the desired τ_2 and that measured by the sensor.

The controller effectively cancels the static error. An overshoot appears, but the measured value quickly tends towards the desired measurement. The *instability* observed in the graphs before reaching the expected value can be due to different factors:

- The effects of pressure ingress and egress in the muscles are related to fluid mechanics. However, these effects have not been modelled.
- The established models are purely static models. There is a certain equilibrium time for shortening when the muscle undergoes a change in pressure. It is not instantaneous, the bandwidth of the actuator is not infinite. The dynamics of pressure change within the muscle depend on two factors. Firstly, the variation is different if the pressure is increasing or decreasing. The pressure increase and decrease are not synchronised in time. Secondly, the pressure drop is non-linear. The intake and exhaust time of the air changes according to the pressure. It is likely that the index response of the muscle to a certain pressure change depends on the initial pressure. A change in pressure when the muscles are both at high or low pressures may have a different impact on stability. This is however completely neglected and not modelled in the controller. The controller gains are constant for the entire system. They do not vary with the initial pressure and are not adaptive. It is then possible that the actual dynamics are not what is expected. If the system includes additional dynamics to those modelled, it can quickly become unstable. Integral action and too high gains can then have a detrimental effect on the system.
- The dynamics of the I/P converters can create a pure delay. Indeed, not all the I/P converters used have the precise same dynamics, so that the muscles do not receive their pressure at the exact same time.

- The length of the tubes used to convey the pressure to the robot have a certain length. This can also create a delay. In addition, the tube of each muscle is not exactly the same length.
- The physics of the muscle may be slightly different between each of the muscles used. However, they are industrially manufactured, so they should behave in the same way.
- The fact that the braid is set in rubber does not eliminate dry friction fibre to fibre. Dry friction introduces a delay. Although the rubbers and their thicknesses are the same, it is possible that the braids and the small springs in the braids generate different positions and therefore different reactions to the swelling.

When the robot is used in a game, it is unlikely that a step command will be requested. Step operation is indeed not desirable regarding interaction with the user. Instead, the force increase is likely to be continuous. In order to observe the effect of the controller in this case, a similar test is performed, where the desired torque is not sent as a step command, but as a continuous sinusoidal command (FIG. 5.4). The command then changes at each time step and varies very slightly. The corrector acts in a very punctual way. The graph of the continuous error is drawn.

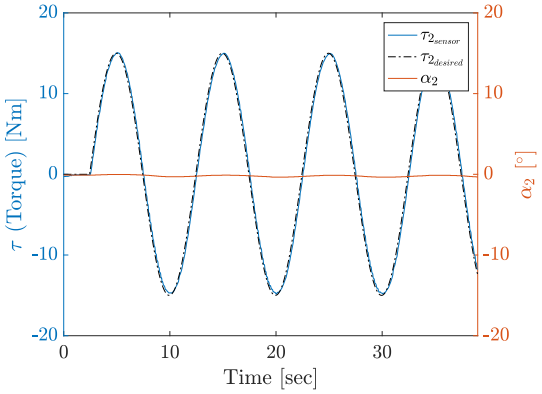


Figure 5.4: Comparison between the desired τ_2 and that measured by the sensor when $\alpha_2 = 0$.

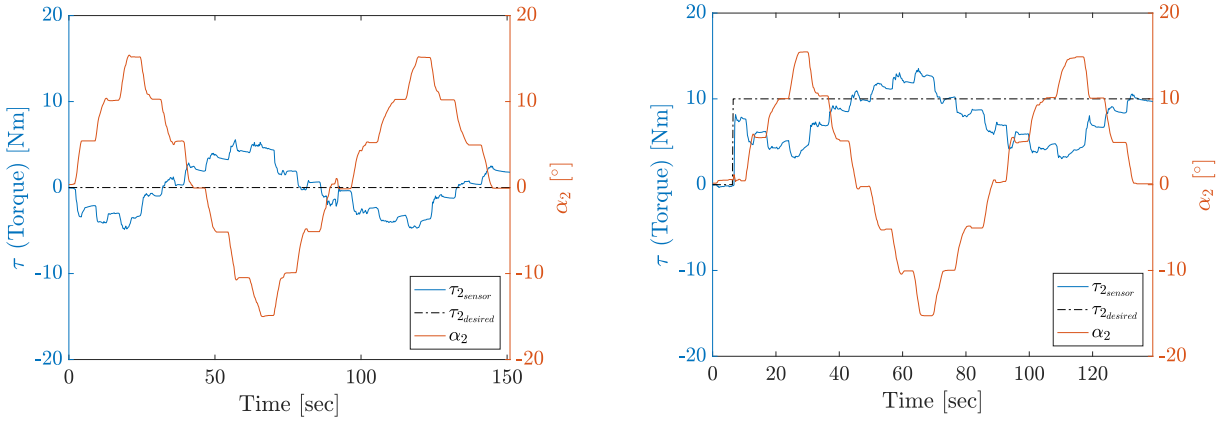
It can be seen that the continuous error is quickly and precisely cancelled out. Improved tuning of the controller gains can probably lead to even better results. This is an investigation for future work on the robot.

5.3 Validation task 2: Constant torque control

Validation task 1 focused on the robot behavior when interfaced with a very rigid load, since the handle was mechanically blocked. The second test assesses the robot's ability to meet another objective: constant torque control. For example, if the system receives a zero-torque command, it must behave transparently when the handle is manipulated. In this specific task, we analyze a case where the handle is completely free and thus when the load impedance is the lowest.

5.3.1 Open loop control

This test is again performed on joint 2 only. This time, the desired joint torque τ_2 is kept constant while the arm position changes, i.e. α_2 varies. Fig. 5.5 shows two tests: when the reference torque τ_2 is zero, and when it is 10 N m. The angle α_2 varies from $-15°$ to $15°$.



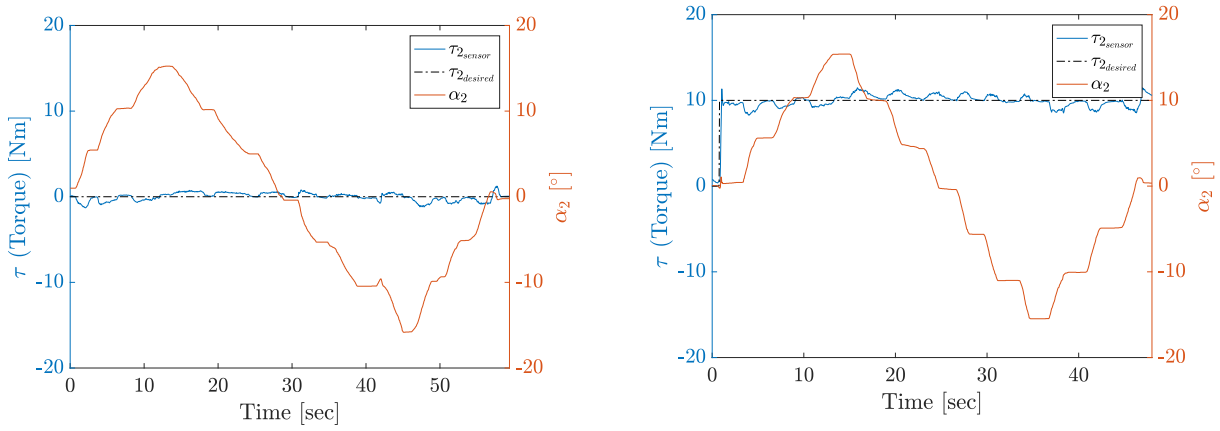
(a) $\tau_2 = 0$ (b) $\tau_2 = 10$

Figure 5.5: Comparison of the desired τ_2 and that measured by the sensor when α_2 varies.

The torque value does not remain constant at the desired value but oscillates around it. Given the large difference in values, the open loop control for this test is not validated. In both cases, it is necessary to close the loop in order to refine the control.

5.3.2 Closed loop control

Identical tests, keeping the torque reference constant, were carried out with the system in closed loop. The results for a torque of 0 and 10 N m are shown in Fig. 5.6.



(a) $\tau_2 = 0$ (b) $\tau_2 = 10$

Figure 5.6: Comparison of the desired τ_2 and that measured by the sensor when α_2 varies.

The results show a clear improvement in control when the system is in closed loop. The measured torque value still varies around the desired value, but with smaller variations. Moreover, when the position is stabilised, the measured value tends towards the desired one.

5.4 Analysis of closed loop system performance

It would be interesting to characterise the ability to control this system in a closed loop system, with the demanding condition of behaving as a zero impedance and therefore zero torque system. We should observe to what extent the robot follows the setpoint if it is subjected to sudden torque variations linked to a change in angular position. When working with a zero torque command, the robot should react as quickly as possible. But the system no longer reacts correctly in a spontaneous way above a certain speed. It then behaves like a system with springs and dampers. The limit depends both on the controller's ability to detect changes quickly and to react to them, but also on mechanical characteristics such as the airflow, which may be too slow. In order to characterise this limit, new experiments are conducted to perform a frequency analysis of the system. Creating the Bode diagram of the system will allow to quantify its bandwidth, and thus quantify the performance of the system in closed loop.

5.4.1 Bode diagram of the impedance

The Bode diagram is the frequency analysis of a linear and time-invariant transfer function, i.e. a transformation of one physical variable into another through a dynamic system. It provides a representation of the frequency responses. It graphically represents the gain in decibels of the transfer function and allows to see the behaviour of a controlled system.

During the task, the robot is asked to behave transparently. A constant torque command is sent and the robot must maintain this value when manually moving the arm. However, the results showed that the controller is not perfect and that an additional torque is opposed to the movement. This is mostly observed when the speed of the movement exceeds a certain threshold. The desired Bode diagram represents the frequency response of the impedance Z. The impedance Z, of unit $\mathrm{N\,m\,rad^{-1}}$, is defined as the transfer function of a velocity input to the opposing torque.

$$Z(s) = \frac{\tau_2}{-v_2} = \frac{\tau_2}{-\dot{\alpha_2}} \tag{5.4}$$

To obtain the Bode diagram of the impedance Z, an organised procedure was followed. First, a test was performed, using the robot. The torque reference is maintained to zero, while the handle is manually stimulated at various frequencies. The aim is to create a frequency-rich signal in order to refine the results observed in the Bode diagram. This will result in a good frequency spectrum of the generated signal. The resulting time signal lasts approximately 13 minutes and is shown in Fig. 5.7.

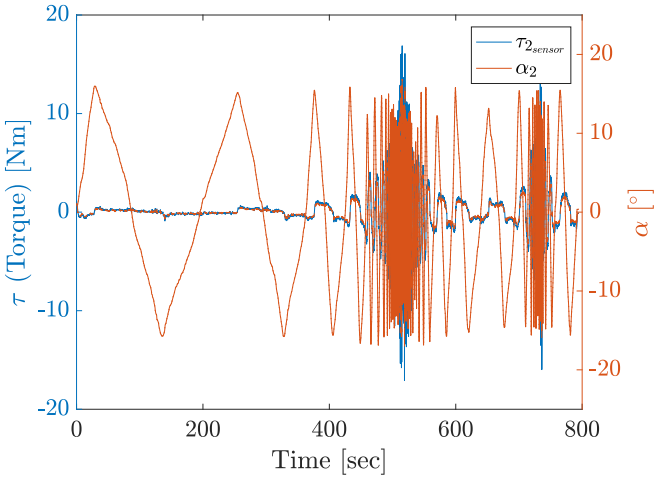
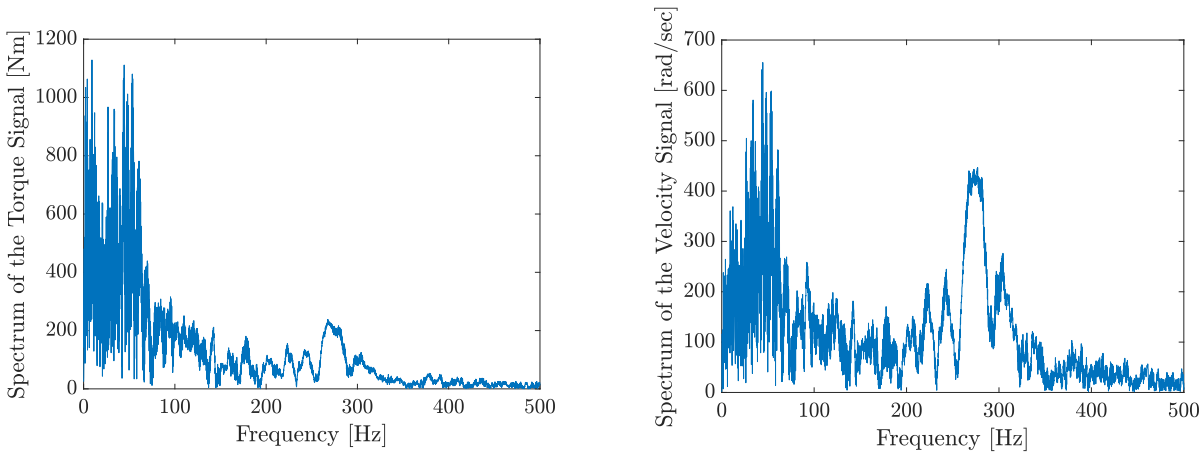


Figure 5.7: Time signal of the test performed.

Then, the frequency signal of the torque as well as the velocity are extracted from the time signal by calculating their Fourier transform, using the Matlab function *fft* (Fig. 5.8).



(a) Torque (b) Velocity

Figure 5.8: Frequency spectrum of signals.

Finally, by taking the ratio of the two frequency spectra, it is possible to create the experimental Bode diagram (Fig. 5.9). It is important to note that the x-scale, representing the frequency, is logarithmic.

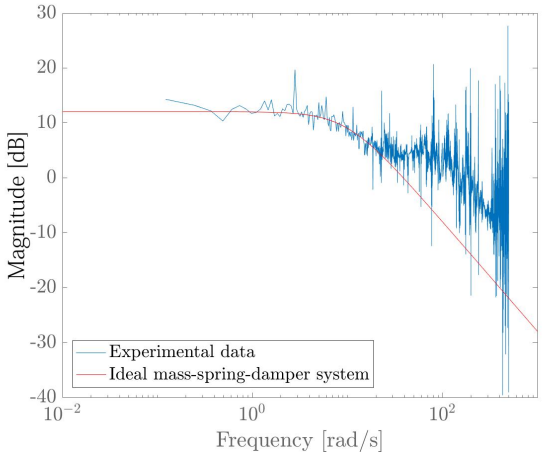


Figure 5.9: Bode diagram of the impedance Z.

5.4.2 Analysis of results

To be able to interpret the Bode diagram of the impedance Z of this system, it is necessary to understand what the different asymptotes of the graph correspond to. To do this, an analogy with the Bode diagram of the impedance Z of an ideal mass-spring-damper system is necessary.

Bode diagram of the impedance Z of the mass-spring-damper model

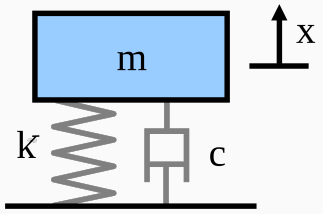


Figure 5.10: Representation of the mass-spring-damper system.

The equation of motion of the mass-spring-damper model is given by the sum of the forces on the mass.

$$F = m\ddot{x} + c\dot{x} + Kx$$

Passing into the Laplace domain and highlighting the velocity parameter, the equation becomes:

$$F = \left(ms + c + \frac{K}{s}\right) V$$

The transfer function of the mechanical impedance is then the following:

$$\begin{aligned} \frac{F}{V} &= ms + c + \frac{K}{s} \\ &= \frac{ms^2 + cs + K}{s} \end{aligned}$$

The Bode diagram of this system is shown in Fig. 5.11.

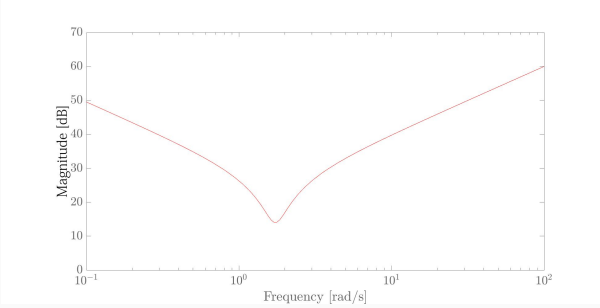


Figure 5.11: Bode diagram of the impedance Z of the mass-spring-damper system.

The transfer function informs us that at low frequencies, the characteristic of a pure spring dominates. An oblique asymptote of slope K is observed which tends to infinity. At high frequencies, the term representing the behaviour of the mass dominates. An oblique asymptote of slope m is observed. And in between, the effect of the damper is apparent. Three categories are identified:

- A *decreasing oblique asymptote* describes the behaviour of a pure *spring*.
- An *increasing oblique asymptote* describes the behaviour of a pure *mass*.
- A *horizontal asymptote* describes the behaviour of a pure *damper*. Horizontal convergence means that there is a direct proportionality between the torque and the speed, like a damper does.

The asymptotic curves of the Bode diagram of the impedance Z (FIG. 5.9) follows each of the three characteristic lines as a function of the dominant frequency. The results show that:

- *At low frequencies*, the system behaves like a damper. It has no residual stiffness as the system is programmed to have zero impedance. The controller manages to cancel the spring effects. There remains the residual friction of the system, such as mechanical friction in the pivots for example, because it is a movement that will oppose a change in speed.
- *At high frequencies*, the system behaves mainly like a spring. When the robot is stimulated at high frequency, it will have a highly elastic behaviour. There are compressibility effects of the fluid in the muscles that actually constitute a spring. This is a speed-dependent phenomenon. The residual stiffness phenomenon dominates the friction. The robot no longer acts transparently. This means that the closed loop control is no longer able to maintain a zero impedance. The slope of the oblique asymptote is proportional to the physical stiffness of the device.
- If even *higher frequencies* were reached, the effect of the mass would surely appear. The accelerations would be so large that it would cost a lot of torque to accelerate the residual mass of the robot. However, the robot is too light and these frequencies are not reached.

The Bode diagram fulfils the primary objective: to characterise the bandwidth of the system achievable in closed loop with the controller gains as tuned. To do this, it is necessary to look at the frequency at which there is a 3 dB loss from the stationary level. The amplitude of the stationary level is 12 dB. When we go down to 9 dB, the frequency is $10\,\mathrm{rad\,s^{-1}}$. So, the experimental bandwidth of the system is 1.59 Hz. Beyond this frequency, the system does not respond correctly to the command. This work gives a limit on the maximum dynamics achievable with this device.

5.4.3 Improvement of results

To increase the bandwidth of the system and thus improve its dynamics, it has been attempted to adjust the gains by trial and error. Finally, the values of the gains chosen are the following:

$$\begin{cases} K_p & = 1 \\ K_i & = 25 \quad \rightarrow \quad \Gamma_i = 0.01 \end{cases} \tag{5.5}$$

By following exactly the same method, the bode diagram of a new signal (Fig. 5.12) generated with the new gains is obtained (Fig. 5.13).

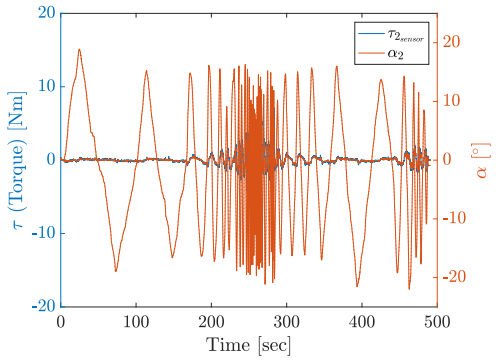


Figure 5.12: Time signal of the test performed.

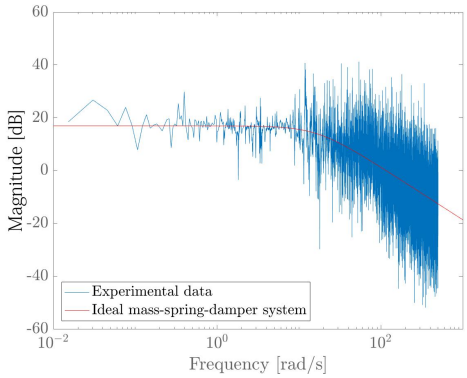


Figure 5.13: Bode diagram of the impedance Z.

The same form of graph is obtained. The horizontal asymptote indicates a damper behaviour at low frequency. The decreasing oblique asymptote demonstrates the effect of the spring at high frequency. The observed bandwidth is 2.66 Hz. The dynamics of the system is thus slightly improved. The values of these gains are kept for further work. Again, a better tuning of the gains will lead to a higher bandwidth.

5.5 Summary

The control of the system is achieved through a combination of two actions: feedback and feed-forward control. Here, the control is applied to the joint torque. The closed loop system corrects the measurement based on the reference torque. The implemented controller is validated through two different tasks. The first one verifies the regulation of a torque that varies when the joint configuration is blocked. The second analyses the opposite case. It checks the regulation of a torque that is kept constant when the joint configuration varies. The results show the reliability of the feedback loop.

The performance of the closed loop system is analysed by creating the Bode diagram of the impedance. It is observed that the system no longer responds to the requested command above a certain frequency, called the cut-off frequency.

The next chapter presents a high-level implementation that shows a typical human-robot interaction. It introduces a new type of control: the hybrid stiffness control, which combines a hardware stiffness provided physically by the pneumatic actuators and a software stiffness implemented mathematically.

Chapter 6

Hybrid stiffness control towards compliant interfaces

In order to highlight a possible interaction between the patient and the robotic arm, it is necessary to implement a high-level task. This chapter first presents the task implemented in this work: a spring. Then, a hybrid stiffness control is considered. This allows us to highlight the major advantage of artificial muscles: their natural compliance. The hybrid stiffness control relies, as far as possible, on the physical stiffness of the robot, and uses software stiffness if necessary. After presenting the block diagram of this control, different tests are performed to analyse the performance of the system. To do this, the Bode digram search is performed again. The objective is to find out if regulating the physical stiffness makes the stiffness control more optimal.

6.1 High-level implementation: the spring

The implemented controller will allow to create a high-level task that highlights a typical human-robot interaction. The physiotherapist's job would be to define, for example, a reference trajectory and a desired stiffness that the robot submits to the user to prevent him from deviating from it. The input parameters are therefore a reference position, α_{ref} and a stiffness, K_{ref}. From there, it is possible to calculate the force and stiffness to be applied to the robot so that the robot's behaviour is the one programmed by the physiotherapist. The high-level task chosen is to ask the robot to behave like a spring. As the handle moves away from a chosen equilibrium position, the user should feel an increasing force trying to bring it back to that equilibrium position. Fig. 6.1 illustrates the result of a simple test involving the action of a spring.

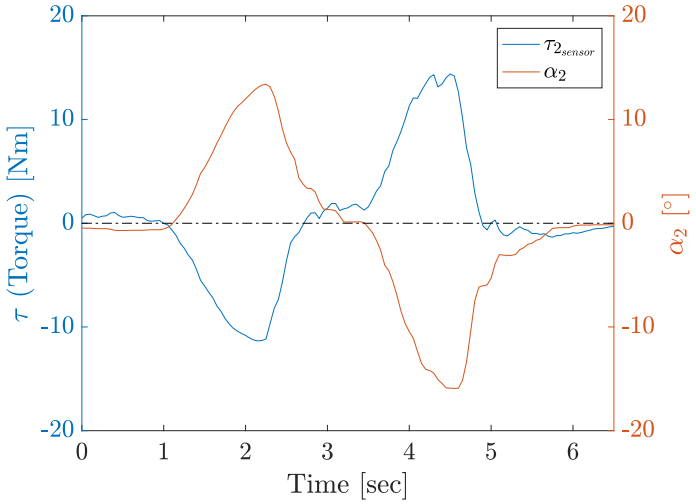


Figure 6.1: Evolution of the torque as a function of the angle of the spring.

We can observe that the more the handle moves towards positive angle values, the more a negative torque value, proportional to the desired stiffness, is felt. And vice versa. The implementation of a simple spring is thus validated. However, given the naturally compliant physics of artificial muscles, it is interesting to ask if it is possible to take advantage of it during the stiffness control.

6.2 Hybrid stiffness control

A hybrid stiffness control is the combination of two types of stiffness: hardware stiffness and software stiffness. In this section, both concepts are explained and a block diagram of this hybrid stiffness control is presented.

6.2.1 Hardware & software stiffness

The stiffness behaviour can be induced in two different ways.

1. ***Hardware/physical stiffness:*** The robot activated by artificial pneumatic muscles has an intrinsic elasticity due to the physics of the system. The spring effect can be induced by the robot hardware. Even though the pressures do not change, an increase or decrease in force is actually felt when the handle is manipulated. The stiffness is felt by real mechanical deformation of a physical element of the system. It is possible to modulate the elasticity of the system so that it behaves physically as a spring.

2. ***Software stiffness:*** The spring effect can be modelled in software. That is, it is possible to implement a force to be applied to the handle that is proportional to the distance between the desired position and the measured position. This requires the implementation of a stiffness law. In other words, specify the desired force and stiffness as a function of an error in position.

The second way is usually used in rigid, electrically driven robot controls, like the REAplan. A stiffness can always be rendered with programming. If it is desired that the handle reacts to a virtual deformation, it is sufficient to control the interface force as if it were a spring. For example, with the REAplan, the stiffness felt at the handle is generated in software only. This robot is in theory infinitely rigid, there are no deformations. It reacts to a force disturbance only via the position of the actuator. However, it is not comfortable and mechanically it is not ideal because there may be problems with latency, noise, resolution, etc.

With the robot used here, it would be advantageous to combine both behaviours. The force felt at the handle would be the result of a physical stiffness of the device and a software-induced stiffness. The physical limits of the system to impose a given force or stiffness becomes an interesting thought. It would be necessary, as far as possible, to rely on the hardware stiffness, and to compensate for this by software stiffness only when it falls outside the achievable range of hardware stiffness. The limits of the physical (hardware) stiffness of the system are known from the model and are shown in Fig. 4.3 of Sec. 4.2. They depend on the position and the desired torque. An important question then arises.

Scientific questioning of hybrid stiffness control

Is it relevant to regulate the physical stiffness of an elastic system in order to optimise a simulated virtual stiffness control at the handle?

To be able to answer this question, it is first necessary to create the block diagram of this hybrid stiffness control.

6.2.2 Bloc diagram of the hybrid stiffness control

For each joint, there are two actuators, and therefore two control variables that can be specified: the force and the equivalent stiffness. A stiffness law can be added to the controller implemented in the previous chapter in order to obtain a control in force and stiffness. Therefore, the control diagram block can be revisited to add stiffness control.

The inputs to the system, i.e. the commands chosen by the user, are:

- An equilibrium position (the joint angle), α_{ref}
- A desired spring stiffness, K_{ref}

In this version of the control, the hardware stiffness value must also be selected. In a future version, it would be interesting to implement a block that would allow to know the optimal value of the hardware stiffness according to the desired stiffness. In this way, the user would only have to select two variables.

As long as the desired stiffness, K_{ref}, can be achieved by the hardware stiffness, no software stiffness is needed, and the system remains in open loop. The inverse model calculates the pressures to be injected into the two muscles according to the equilibrium position, the hardware stiffness and the desired torque. When the hardware stiffness is no longer sufficient, i.e. when it is no longer equal to the desired stiffness, a software stiffness, and therefore a software torque, is added. This modifies the pressures calculated by the inverse model. The software torque is passed through the control loop in order to have a measured torque that tends towards the desired torque. The final diagram of the hybrid stiffness control is shown in FIG. 6.2.

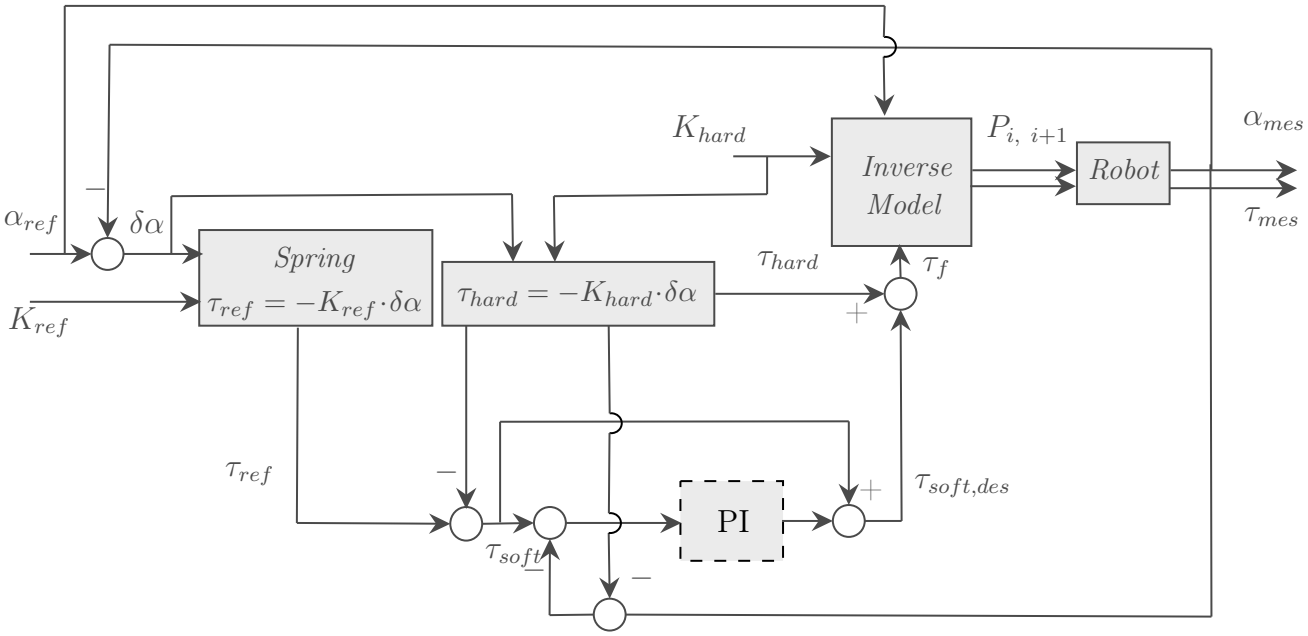


Figure 6.2: Block Diagram of the hybrid stiffness control.

To better understand this block diagram, let's use a numerical example:

Let us assume that the equilibrium position is 0 rad and the desired stiffness is $40\,\mathrm{N\,m\,rad^{-1}}$. If the handle is moved at an angle of 0.2 rad, the desired torque, τ_{ref}, will be $-8\,\mathrm{N\,m}$.

- In the *first case*, the hardware stiffness is equal to the desired stiffness: $K_{ref} = K_{hard}$. The hardware torque is also $-8\,\mathrm{N\,m}$. τ_{soft}, which is $\tau_{ref} - \tau_{hard}$, will then be $0\,\mathrm{N\,m}$. If the system is considered to be perfect, a value of zero enters the PI controller, as the hardware torque is subtracted from the measured torque. $\tau_{soft,des}$ is also $0\,\mathrm{N\,m}$ and the feedback loop is not used. The value of τ_{hard} is added to the input of the inverse model. The pressures to be supplied to the muscles are calculated based on $\alpha_{ref} = 0°$, $K_{hard} = 40\,\mathrm{N\,m\,rad^{-1}}$ and $\tau_f = \tau_{hard} + \tau_{soft} = -8\,\mathrm{N\,m} + 0\,\mathrm{N\,m} = -8\,\mathrm{N\,m} = \tau_{ref}$.

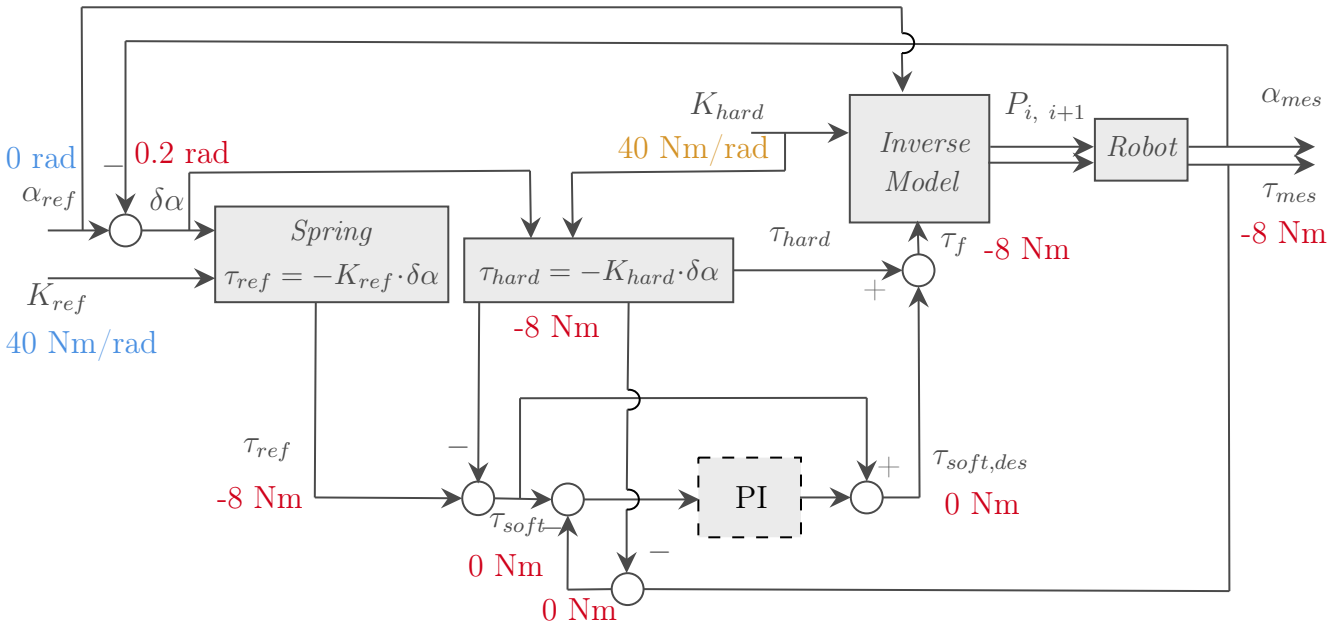


Figure 6.3: First example of the hybrid stiffness control.

- In a *second case*, the hardware stiffness is no longer sufficient, it is lower than the desired stiffness: $K_{ref} > K_{hard}$. If the hardware stiffness is $10\,\mathrm{N\,m\,rad^{-1}}$, the hardware torque is $-2\,\mathrm{N\,m}$. τ_{soft} is then $\tau_{ref} - \tau_{hard} = -8\,\mathrm{N\,m} - (-2\,\mathrm{N\,m}) = -6\,\mathrm{N\,m}$. Again, assuming the system is perfect, the measured torque value is $-8\,\mathrm{N\,m}$. To this value, τ_{hard} is subtracted, which gives $-6\,\mathrm{N\,m}$. A value of $0\,\mathrm{N\,m}$ enters the PI controller, and the feed-forward action adds the $-6\,\mathrm{N\,m}$ to the output. Finally, a torque $\tau_f = \tau_{hard} + \tau_{soft} = -2\,\mathrm{N\,m} + (-6\,\mathrm{N\,m}) = -8\,\mathrm{N\,m} = \tau_{ref}$ is fed into the inverse model. This time the feedback loop is used.

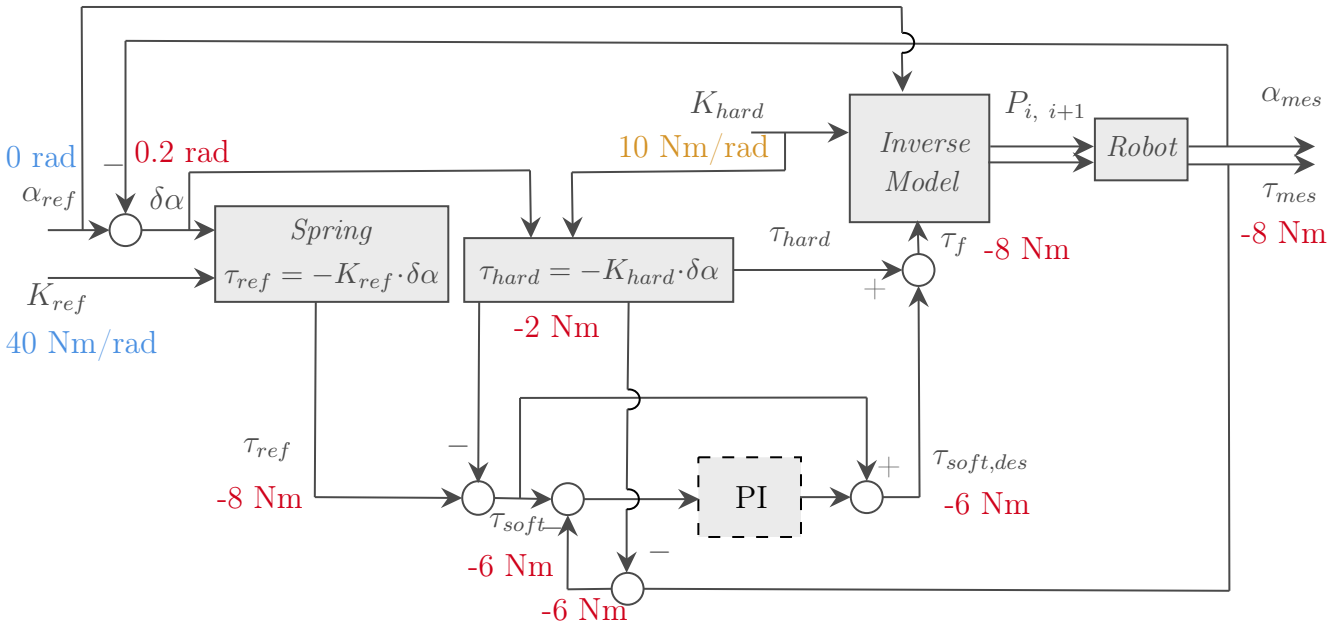


Figure 6.4: Second example of the hybrid stiffness control.

6.3 Validation task 3: Stiffness control

To verify the correct functioning of the hybrid stiffness control, three cases are analysed.

3 cases to analyse the performance of hybrid control

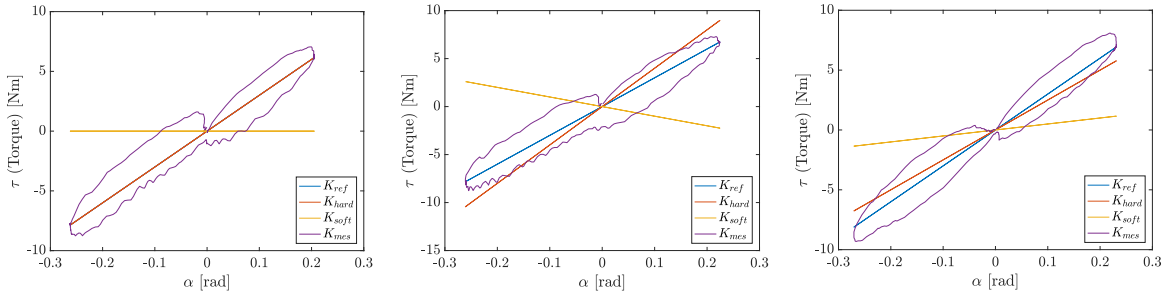
1. $K_{ref} = K_{hard}$: The desired stiffness is equal to the hardware stiffness. There is therefore no need to use the software stiffness.
2. $K_{ref} < K_{hard}$: The desired stiffness is lower than the hardware stiffness. Software stiffness must be added negatively to achieve the desired stiffness.
3. $K_{ref} > K_{hard}$: The desired stiffness is greater than the hardware stiffness. Software stiffness must be added positively to achieve the desired stiffness.

In the first case, the stiffness is the closest to the virtual spring that we want to implement. This means that the robot will, in open loop, already generate a resistance torque that will be equal to the desired one. The software loop does nothing if the model is perfect. In the last two cases, the physical stiffness is not equal to the desired one. The implementation of the spring will have to rely on the software stiffness. This means that part of the stiffness implementation will be based on the stiffness control loop.

So, to analyse those three cases, a third task is conducted. Again, only the second joint is used, the first being mechanically blocked. The robot is implemented to act as a spring. An equilibrium position is chosen ($\alpha_2 = 0°$), as well as the desired stiffness and the hardware stiffness. Their values for each experiment are shown in TAB. 6.1. The handle is moved manually from approximately 15° to −15°. FIG. 6.5 shows the results of the three experiments, where the different stiffnesses are expressed in $\mathrm{N\,m\,rad^{-1}}$.

	Test 1	**Test 2**	**Test 3**	**Unit**
K_{ref}	30	30	30	$\mathrm{N\,m\,rad^{-1}}$
K_{hard}	30	40	25	$\mathrm{N\,m\,rad^{-1}}$

Table 6.1: Stiffness values for each test.



(a) $K_{ref} = 30 = K_{hard}$ (b) $K_{ref} = 30 < K_{hard} = 40$ (c) $K_{ref} = 30 > K_{hard} = 25$

Figure 6.5: Stiffnesses for the three experiments.

The results correspond to the expectations. The software stiffness adds positively, negatively or not depending on the value of the desired stiffness compared to the hardware stiffness. The measured stiffness has the shape of an ellipse due to the hysteresis of the muscles. However, this ellipse is correctly situated around the desired stiffness.

6.4 PERFORMANCE ANALYSIS

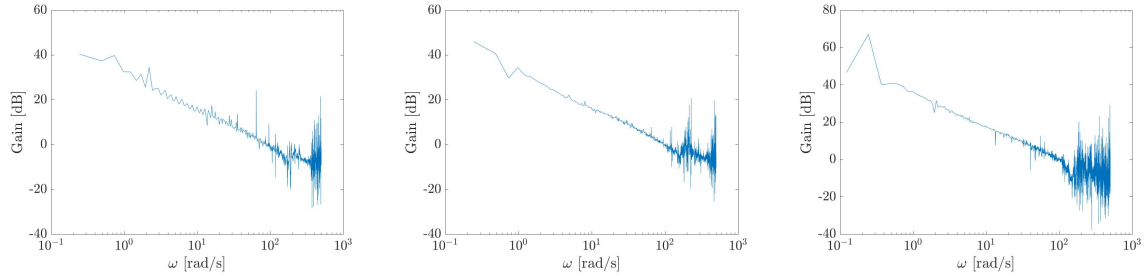
In order to know if regulating the physical stiffness of the system according to the desired stiffness optimises the stiffness control, it is interesting to discuss the difference in terms of control load for the different cases. The expected result would be that the impedance load of the closed loop is much lower when the hardware stiffness is close to the desired value. To validate or not this intuition, it is useful to plot the Bode diagram of each of the three cases in order to know the bandwidth of each situation.

6.4.1 Bode diagram of the impedance

To create the bode diagram, a procedure similar to that performed in Sec. 5.4.1 is followed. A time signal, rich in different frequencies, was created by moving the handle manually at different speeds. The robot was again asked to behave like a spring around a certain equilibrium position. Different tests were performed for different values of desired stiffness and hardware stiffness. They are shown in Tab. 6.2. From the time signal of the measured torque and the joint angle, the Fourier transforms of the torque and velocity signal are obtained. Then, the Bode diagram of the impedance $Z = \frac{\tau_2}{-\dot{\alpha}_2}$ is produced. The obtained Bode diagrams are shown in Fig. 6.6.

	Test 1	**Test 2**	**Test 3**	**Unit**
K_{ref}	30	30	40	$\mathrm{N\,m\,rad^{-1}}$
K_{hard}	30	40	25	$\mathrm{N\,m\,rad^{-1}}$

Table 6.2: Stiffness values for each test.



(a) $K_{ref} = 30 = K_{hard}$ (b) $K_{ref} = 30 < K_{hard} = 40$ (c) $K_{ref} = 40 > K_{hard} = 25$

Figure 6.6: Bode diagram for the three experiments.

6.4.2 Analysis of results

The decreasing oblique asymptote represents the behaviour of a spring. At low frequencies, the system therefore behaves as desired. The horizontal asymptote describes the behaviour of a damper. At high frequencies, the system no longer responds to the command. From a certain frequency, the cut-off frequency, the system no longer behaves like a spring, but becomes a damped system. The value of this frequency can be determined by looking at the Bode diagram. This is the point where the oblique asymptote becomes a horizontal asymptote. The values of the cut-off frequencies for the three cases are shown in Tab. 6.3.

	Test 1	**Test 2**	**Test 3**
	$K_{ref} = K_{hard}$	$K_{ref} < K_{hard}$	$K_{ref} > K_{hard}$
f_c	30.2 Hz	27.06 Hz	24.67 Hz

Table 6.3: Cut-off frequency for each test.

Although the largest bandwidth is observed in the case where the desired stiffness is equal to the hardware stiffness, the cut-off frequencies are not fundamentally different. They revolve around the same range of values. The relevance of regulating the physical stiffness in order to optimise the stiffness control is therefore questionable. Nevertheless, the results are encouraging. Further research of this hybrid control would lead to a new type of interesting and fully exploitable stiffness control.

6.5 Summary

A high-level task is implemented. It consists in requesting the robot to behave like a spring, with a certain desired stiffness, around a point of equilibrium. This control can be realized in a hybrid way, i.e. combining hardware stiffness and software stiffness. The hardware stiffness is intrinsic to the physics of pneumatic muscles. The software stiffness is implemented via the software. Hybrid stiffness control takes advantage of the pneumatic actuators by exploiting the physical stiffness to the maximum, and using software stiffness only when necessary. When characterizing the performance of the system via the Bode diagram, we observe that the bandwidth is greater when the desired stiffness can be fully achieved by the hardware stiffness. Therefore, regulating the physical stiffness of an elastic system allows to optimize the control in stiffness at the robot handle.

Chapter 7

Conclusion & Perspectives

This master thesis is a contribution to an innovative project in the field of medical robotics: the creation of a robotic arm activated by artificial pneumatic muscles for post-stroke rehabilitation of the upper limb. This work allowed to better understand the very specific physics of these activators, as well as the functioning of the robot. The main stages of the work are summarised in this chapter, along with the results and key points to be retained. Then, a perspective of the project is given. Avenues for the continuation of the work are proposed, as well as ways to improve the mechanics of the arm.

7.1 Conclusion

Firstly, a state of the art in the field of artificial pneumatic muscles used in robotic therapy has shown that the development of such actuators deserves to be addressed. The physics of the muscles makes them particularly soft, giving them a behaviour similar to that of human muscles. This natural compliance is a significant advantage and makes the robot safe to use. This type of actuators, besides being cheap, is also strong and light, and has a good power/weight ratio. However, although much research is being done on the introduction of PAMs into medical robotics, there are no commercialized prototypes yet. These actuators have certain disadvantages that hinder their arrival on the market. These include less easy and less precise control than with an electric motor, a limited operating range or loud noise during use due to the compressed air supply.

After a brief description of the robot and its system, the muscle model had to be found. By inverting the model, it was then possible to know the pressures to be sent to each muscle in order to feel a desired force or stiffness in a certain position. A theoretical model exists for the McKibben muscles. However, the Festo muscles, used for the robot, are thicker and have a different behaviour. Therefore, they do not fit into the thin-walled theory used to create the theoretical model. As a result, an experimental, purely empirical model was created, in

spite of any theory. This new model takes into account 11 parameters, which is justified by sufficient accuracy. The pressure parameter P was kept at first order so that the model was analytically invertible. The muscle model implemented in the software is as follows:

$$F(P,\varepsilon) = aP\varepsilon^4 + bP\varepsilon^3 + cP\varepsilon^2 + dP\varepsilon + eP + f\varepsilon^5 + g\varepsilon^4 + h\varepsilon^3 + i\varepsilon^2 + j\varepsilon + k$$

The robot is intended to be used in rehabilitation therapy, especially as a game to motivate the patient. The physiotherapist has to know the limits of the system in order to ask for tasks that can be achieved by the robot. Using MatLab codes, it was possible to determine the maximum force achievable by the arm in all directions of the workspace. The value found is 27 N. The same work was done for the stiffness. It depends on the force and the position of the arm. The values of the minimum and maximum joint stiffnesses found, for zero torque, are listed in Tab. 4.1. At zero torque, the system can provide a maximum stiffness of $117.7\,\mathrm{N\,m\,rad^{-1}}$ with joint 1 and $50.31\,\mathrm{N\,m\,rad^{-1}}$ with joint 2 in all directions of the domain. However, the values are too different depending on the configuration of the arm. Giving the user a single value greatly reduces the usable potential of the robot.

Then, the system low-level control was implemented. A feed-forward action and a feedback action, via a PI controller, are used. The gains of the controller were found mathematically. Several tests, performed on the second joint only, validated the effectiveness of the controller. In the first test, the torque command was varied, while the angle was kept constant. The results proved that the controller works well because the static error, present in the open loop system, was eliminated. The second test consisted of making the robot transparent, i.e. asking it to make the handle feel zero torque when it is moved manually. Again, the results were improved with the presence of the controller. However, when the handle was excited at too high a speed, the behaviour no longer respected the input command. The Bode diagram of the system impedance allowed to identify the bandwidth of the system, which is 2.66 Hz.

Finally, a high-level task, highlighting human-robot interaction, was implemented. This is necessary as the device has to be entered into the clinical context. The robot was asked to behave like a spring around a desired position, with a desired stiffness. An attempt was made to create a hybrid stiffness control. This consists in exploiting the physical stiffness of the device as much as possible, adding software stiffness only if needed. Again, the arm stopped behaving like a spring above a certain speed. The impedance Bode diagram was once again made to characterise the bandwidth of the system. This is 30 Hz when the physical stiffness is equal to the desired stiffness. This value is slightly lower when a software stiffness is required. It can therefore be interesting to regulate the physical stiffness in order to optimise the stiffness control.

Artificial pneumatic muscles are actuators that may have a future in the world of robotic therapy. The results prove that the control is satisfying, especially for the tasks requested in this kind of therapy. Better tuning of the controller gains can improve the system bandwidth. Further research could also lead to a better exploitation of the physical stiffness of the system. Overall, this project shows encouraging results and deserves to be continued.

7.2 Future work & Areas for improvement

As you might expect, the work does not stop here. This last sub-chapter aims to give some ideas for improving the system, in order for the project to continue.

The control can be extended to 2 DoFs by making the first joint free. This would require doubling the control block of the second joint. If the torque and stiffness control works for each joint, it is then possible to switch to the handle frame via the Jacobian operator. This would achieve the ultimate goal of the control: to know the pressures to be sent to each muscle according to a desired force or position, and a desired stiffness felt at the handle.

However, before releasing the first joint, a new version of the mechanical design must be made. Indeed, during the activation of the arm, some mechanical weaknesses could be highlighted. These need to be improved before the full control of the system can be completed. Here is a list of the problems observed and possible improvements to solve them.

- The handle deforms slightly when under tension. The tensile stiffness of the handle should be increased by changing the material or features used in the 3D printing process.
- The space to hold the handle is a bit too narrow. The back of the patient's hand touches the attachment with the segment in some configurations. It would be necessary to add a few extra centimeters to the place reserved for the user's hand. The forearm must then be shortened in the same way to keep the same final length of the segment.
- The pulley of joint 2 bends when the pressure in the muscles is high and the cable is put under great tension. This creates friction with the top piece. The model and the results are distorted. The ball bearing of the joint should be placed higher on the axis. This would create a higher contact point that would compensate for the force of the cable.
- The ball bearing below the handle is not an ideal solution. It creates friction in contact with the table which is not negligible. An alternative solution can be found.
- The angular sensor of the second joint is in contact with the table. This causes significant friction and can wear the sensor. To remedy this, an imbalance has been voluntarily created when fixing the frame to the table. Three additional washers were added to both ends of the base near the arm to raise it. This compensates for the tilt induced by the weight of the arm and the sensor does not touch the table. However, another solution should be considered that would be cleaner and last forever. One solution would be to add a carrier ball at the elbow. Again, some friction will appear. Also, care must be taken that the ball does not come out of the table in any configuration of the arm. Another solution is to place the sensor upwards. To do this, the arm should be mounted in the other direction, and the height of the handle should be adjusted.
- If the joint torque is too high, the cable moves in the pulley. This will disturb the whole system. This situation must be avoided at all costs. One solution would be

to add clamping points between the two parts that surround the cable. If this is not sufficient, an anti-slip material can be added between the contact surface and the cable.

- The pulley cap does not fully conform to the shape of the pulleys for mechanical reasons: the cap cannot touch the segment when the arm is in stop. This shape of the cap does not allow to keep the cable in the pulley permanently. To solve this problem, a triangulation of the part has been made for the first joint. The same should be done for the second joint.

This list is not exhaustive. It can be completed or improved. The proposed solutions are tracks, but any other idea is welcome to ensure a second version of the prototype as robust as possible.

Epilogue

This semester was special and unlike any other semester in my 5 years of study. Working on this thesis was far from being a chore and taught me many things.

I had the chance to work on a project that required collaboration with the medical world. My research on neurorehabilitation taught me new and interesting concepts that I did not previously know. Medical robotics is a field that fascinates me because it gives meaning to the engineering profession. It adds a human dimension to this profession and allows us to put our skills at the service of others. Moreover, I sincerely believe that it is one of the only fields where progress should not stop.

Working on this project was a very good experience. I learned a lot while working in a pleasant and serene environment. I liked the experimental dimension of this thesis. Despite the feeling of not progressing sometimes, I really enjoyed searching, experimenting, and understanding little by little, by manipulating the robot, the very particular physics behind the pneumatic artificial muscles. These few months made me realize how important research is to progress and innovation.

During the semester, I tried to stay as organized as possible. I started using the *Agile* method, which consists in splitting the different objectives into subtasks, and in organizing meetings at regular intervals. The weekly meetings were very helpful throughout the semester and helped me achieve the goals I set in January. There is still work to be done, of course. But I'm very proud of where I've gotten to.

Although I have had moments of doubt, I think this project can really have a future. I am curious to see what the experimental phase with patients will bring. I am firmly convinced that this robot can find its place on the market. I will follow with pleasure the evolution of this project, of which I have added a small contribution. Or at least, time, energy and dedication.

Bibliography

[1] R. Neumann A. Hartmann A. Hildebrandt, O. Sawodny. “Cascaded control concept of a robot with two degrees of freedom driven by four artificial pneumatic muscle actuators”. *American Control Conference*, pages 680–685, 2005.

[2] G. Andrikopoulos, G. Nikolakopoulos, and S. Manesis. "A Survey on applications of Pneumatic Artificial Muscles". *19th Mediterranean Conference on Control and Automation*, pages 1439–1446, 2011.

[3] Axinesis. "REAplan". 2016.

[4] L. Brewer, F. Horgan, A. Hickey, and D. Williams. “Stroke rehabilitation: Recent advances and future therapies”. *QJM: monthly journal of the Association of Physicians*, pages 11–25, Vol. 106, N°1, 2012.

[5] D. Caldwell, N.G. Tsagarakis, S. Kousidou, N. Costa, and I. Sarakoglou. “‘Soft’ exoskeletons for upper and lower body rehabilitation - Design, Control and Testing”. *International Journal of Humanoid Robotics*, pages 549–573, Vol. 3, N°4, 2007.

[6] F. Daerden and D. Lefeber. "Pneumatic Artificial Muscles: actuators for robotics and automation". *Vrije Universiteit Brussel, Department of Mechanical Engineering*.

[7] D. Dochain. "LINMA 1510 Commande linéaire des systèmes dynamique". *EPL*, 2019.

[8] Emotiv. "Neuroplasticity". *Emotiv*, 2020.

[9] C.E. English and D.L. Russell. "Representations of multi-joint stiffness for prosthetic limb design". *Mechanism and Machine Theory*, 2006.

[10] Festo. "Fluidic Muscle DMSP/MAS". *Datasheet*, 2008.

[11] A. Vilchis Gonzales, P. Cinquin, J. Troccaz, and al. “TER: A system for robotic teleechography”. *4th Int. Conf. on Medical Image Computing and Computer-Assisted Intervention, Utrecht, The Netherlands*, pages 326–334, 2001.

[12] J. He, E.J. Koeneman, R.S. Schultz, H. Huang, J. Wanberg, D.E. Herring, T. Sugar, R. Herman, and J.B. Koeneman. "Design of a robotic upper extremity repetitive therapy device". *9th International Conference on Rehabilitation Robotics*, pages 95–98, 2005.

[13] Heidenhain. "Rotary Encoders". *Datasheet*, 2017.

[14] J. Hidler, D. Nichols, M. Pellicio, and K. Brady. “Advances in the understanding and treatment of stroke impairment using robotic devices”. *Topics in Stroke Rehabilitation*, pages 22–35, Vol. 12, N°2, 2005.

[15] Hocoma. "Armeo Power". 2016.

[16] National Instrument. "NI 9265 Datasheet". *National Instrument Corp.*, 2015.

[17] National Instrument. "NI 9403 Datasheet". *National Instrument Corp.*, 2015.

[18] National Instrument. "NI 9215 Datasheet". *National Instrument Corp.*, 2016.

[19] National Instrument. "Systèmes CompactRIO". *National Instrument Corp.*, 2021.

[20] J. Kaiser. "Neurorehabilitation using plasticity and growth as a therapeutic tool". *Burke Neurological Institute*, 2020.

[21] H. Kobayashi and K. Hiramatsu. "Development of muscle suit for upper limb". *IEEE International Conference on Robotics and Automation*, pages 2480–2485, Vol. 3, 2004.

[22] G. Kwakkel, R.C. Wagenaar, G.J. Lankhorst T.W. Koelman, and J.C. Koetsier. “Effects of robot-assisted therapy on upper limb recovery after stroke: A systematic review”. *Neuro-rehabilitation and Neural Repair*, pages 111–121, Vol. 22, N°2, 20.

[23] P. Leconte, J.-J. Obran de Xivry, G. Stoquart, T. Lejeune, and R. Ronsse. “Rythmic arm movements are less affected than discrete ones after a stroke”. *Experimental Brain Research*, pages 1403–1417, Vol. 234, N°6, 2016.

[24] P. Leconte and R. Ronsse. “Performance–based robotic assistance during rhythmic arm exercises”. *Journal of NeuroEngineering and Rehabilitation*, pages (16 pages), Vol. 13, N°82, 2016.

[25] P. Leconte, G. Stoquart, T. Lejeune, and R. Ronsse. “Rythmic upper-limb movement training enhances motor skills of both rhythmic and discrete movements after stoke: A longitudinal study”. *International Journal of Rehabilitation Research*, pages (30 pages), Vol. 1, N°1, 2018.

[26] P.S. Lum, C.G. Burgar, P.C. Shor, M. Maimundar, and M. Van der Loos. “Robot-assisted movement training compared with conventional therapy techniques for the rehabilitation of upper-limb motor function after stroke”. *Archives of Physical Medecine and Rehabilitation*, pages 952–959, Vol. 83, N°7, 2002.

[27] R. Niiyama, S. Nishikawa, and Y. Kuniyoshi. "Athlete Robot". *The University of Tokyo*, 2020.

[28] R. Ronsse. "LMECA2755 Industrial automation". *EPL*, 2019.

[29] V. Squeri, L. Masia, P. Giannoni, G. Sandini, and P. Morasso. “Wrist rehabilitation in chronic stroke patients by means of adaptive, progressive robot-aided therapy”. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, pages 312–325, Vol. 22, N°2, 2014.

[30] B. Tondu. “Towards an efficient inverse static model of a festo actuator made of two antagonist muscles for hybrid control of its position and stiffness”. *Departement de Genie Electrique de l'INSA de Toulouse and LAAS/CNRS in Toulouse*, 2021.

[31] B. Tondu, A. Daidie, S. Ippolito, and J. Guiochet. “A Seven-degrees-of-freedom Robot-arm driven by Pneumatic Artificial Muscles for Humanoid Robots”. *International Journal of Robotics Research*, pages 257–274, Vol. 24, N°4, 2005.

[32] B. Tondu and P. Lopez. “Modeling and Control of McKibben Artificial Muscle Robot Actuators”. *IEEE Control Systems Magazine*, pages 15–38, Vol. 20, N°2, 2000.

[33] T. Trueslen, R. Bonita B. Piechowski-Jozwiak, and al. “Stroke incidence and prevalence in Europe: A review of available data”. *European Journal of Neurology*, pages 581–598, Vol. 13, N°6, 2006.

[34] Wikipedia. "Plasticité neuronale". *Wikipedia*, 2020.

[35] Wikipedia. "Plant (control theory)". *Wikipedia*, 2021.

[36] S. Wongsiri and S. Laksanacharoen. "Design and construction of an artificial limb driven by artificial muscles for amputees". *International Conference on Energy and the Environment*, 2003.

[37] WPI. "Une introduction aux PIDs". *FIRST Robotics Competition*, 2021.

Appendices

Appendix A

Mathematical development to find the physical model of the McKibben muscle.

In the literature of McKibben's muscles, there is the expression of the force F developed by each muscle as a function of the control pressure P and the contraction ratio ε [32] [31].

$$F(\varepsilon, P) = (\pi r_0^2)\ P\ \left[a(1-\varepsilon)^2 - b\right] \qquad \text{with} \qquad \begin{cases} \varepsilon & = \frac{(l_0 - l)}{l_0} \\ a & = \frac{3}{\tan^2(\alpha_0)} \\ b & = \frac{1}{\sin^2(\alpha_0)} \end{cases}$$

where l_0 is the initial length of the muscle, α_0 is the initial muscle braid angle, and r_0 is the initial muscle radius.

As explained in [32], this model is obtained using the principle of virtual work, which can be expressed as follows:

$$P\delta V = -F\delta L \tag{A.1}$$

where δL and δV are respectively the elementary muscle length and volume variation.

If the pressure force is divided into lateral and axial pressure, the virtual work theorem becomes:

$$\delta W_{lateral\ pressure} + \delta W_{axial\ pressure} + \delta W_{equilibrium\ force} = 0 \tag{A.2}$$

$$(2\pi r l P)(+\delta r) - (\pi r^2 P)(-\delta L) - F(-\delta L) = 0 \tag{A.3}$$

From the elementary pantograph opening principle, relations relative to the braided

shell are given:

$$\frac{l}{l_0} = \frac{\cos(\alpha)}{\cos(\alpha_0)} \tag{A.4}$$

$$\frac{r}{r_0} = \frac{\sin(\alpha)}{\sin(\alpha_0)} \tag{A.5}$$

From this, an expression for the radius of the muscle r is derived:

$$r = r_0 \left[\frac{\sqrt{1 - \cos(\alpha_0)^2 \left(\frac{l}{l_0}\right)^2}}{\sin(\alpha_0)} \right] \tag{A.6}$$

Finally, by replacing this expression in Eq. (A.3), the model of force as a function of pressure and contraction ration is found.

Appendix B

Jacobian operator of a SCARA robot with 2 DoFs.

The Jacobian operator is obtained from the direct geometric model. The latter is found via the Denavit-Hartenberg convention and is represented as a transformation matrix from the end-effector frame to the base frame. The matrix is the result of a product of four basic transformations (Fig. B.1): a rotation by an angle θ_i around the z_{i-1}-axis, a translation by a distance d_i along the z_{i-1}-axis, a rotation by an angle α_i around the x_i-axis, and a translation by a distance a_i along the x_i-axis.

$$^{i-1}H_i = Rot(z, \theta_i)\ Trans(z, d_i)\ Trans(x, a_i)\ Rot(x, \alpha_i) \tag{B.1}$$

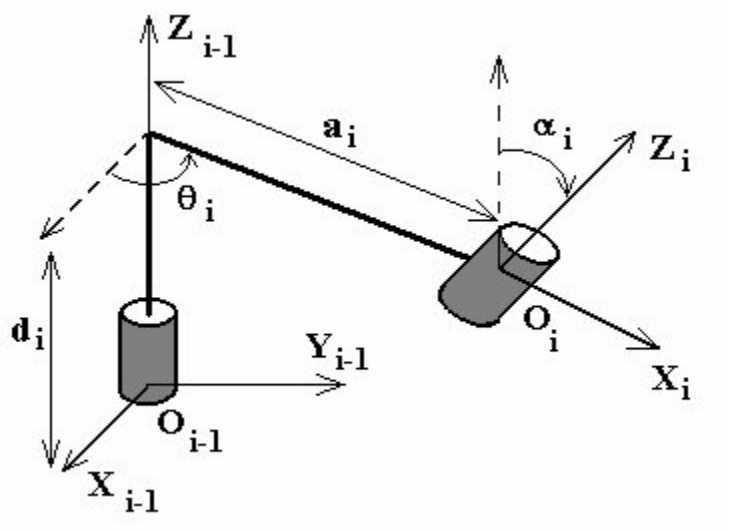


Figure B.1: Denavit-Hartenberg convention.

These transformations result in a transition matrix[1]:

$$ {}^{i-1}H_i = \begin{pmatrix} C_{\theta_i} & -S_{\theta_i}C_{\alpha_i} & S_{\theta_i}S_{\alpha_i} & a_iC_{\theta_i} \\ S_{\theta_i} & C_{\theta_i}C_{\alpha_i} & -C_{\theta_i}S_{\alpha_i} & a_iS_{\theta_i} \\ 0 & S_{\alpha_i} & C_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{B.2} $$

To place the reference frames and variables according to the convention on the robot schematic, some conditions must be satisfied [28].

- The axis x_{i+1} is perpendicular to the axis z_i.
- The axis x_{i+1} intersects the axis z_i.
- The axis z_i is the axis of actuation for joint $i+1$, i.e. the axis of revolution for a revolute joint, and the axis of translation for a prismatic joint.
- a is the distance between z_i and z_{i+1}, measured along the axis x_{i+1}.
- α is the angle between z_i and z_{i+1}, measured in a plane normal to x_{i+1}.
- d is the distance from o_i to the intersection of x_{i+1} and z_i.
- θ is the angle from x_i to x_{i+1}, measured in a plane normal to z_i.

The SCARA with 2 DoFs is shown in Fig. B.2. Looking at this representation and following the convention, we find the values of the variables used to create the Denavit-Hartenberg matrix. They are listed in Tab. B.3.

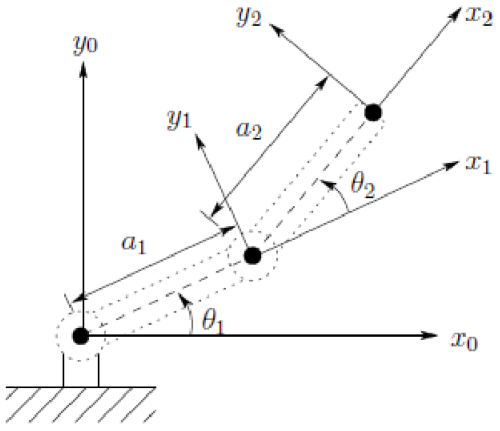


Figure B.2: 2 DoFs SCARA robot.

Link	θ	d	a	α
1	θ_1*	0	a_1	0
2	θ_2*	0	a_2	0

Figure B.3: values of the variables used to create the Denavit-Hartenberg matrix.

[1] C is an abbreviation of *cosine*, and S is an abbreviation of *sinus*.

The transition matrices between references 0 and 1, and between references 1 and 2 are found.

$$ {}^{0}H_1 = \begin{pmatrix} C_{\theta_1} & -S_{\theta_1}C_{\alpha_1} & S_{\theta_1}S_{\alpha_1} & a_1C_{\theta_1} \\ S_{\theta_1} & C_{\theta_1}C_{\alpha_1} & -C_{\theta_1}S_{\alpha_1} & a_1S_{\theta_1} \\ 0 & S_{\alpha_1} & C_{\alpha_1} & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{B.3} $$

$$ {}^{1}H_2 = \begin{pmatrix} C_{\theta_2} & -S_{\theta_2}C_{\alpha_2} & S_{\theta_2}S_{\alpha_2} & a_2C_{\theta_2} \\ S_{\theta_2} & C_{\theta_2}C_{\alpha_2} & -C_{\theta_2}S_{\alpha_2} & a_2S_{\theta_2} \\ 0 & S_{\alpha_2} & C_{\alpha_2} & d_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{B.4} $$

Then, the matrix of passage between the end-effector frame and the base frame is identified.

$$ {}^{0}H_2 = {}^{0}H_1 \cdot {}^{1}H_2 = \begin{pmatrix} C_{12} & -S_{12} & 0 & a_2C_{12} + a_1C_1 \\ S_{12} & C_{12} & 0 & a_2S_{12} + a_1S_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{B.5} $$

where

$$ \begin{cases} C_{12} & = \cos\theta_1 \cos\theta_2 - \sin\theta_1 \sin\theta_2 \\ S_{12} & = \cos\theta_1 \sin\theta_2 - \sin\theta_1 \cos\theta_2 \end{cases} $$

By identification with a transformation matrix, we obtain the rotation matrix, ${}^{0}R_2$, where the last column represents the direction of the z axis, and the translation vector, ${}^{0}O_2$, which represents the position (coordinates at the origins) of the joint.

$$ {}^{0}R_2 = \begin{pmatrix} C_{12} & -S_{12} & 0 \\ S_{12} & C_{12} & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{B.6} $$

$$ {}^{0}O_2 = \begin{pmatrix} a_2C_{12} + a_1C_1 \\ a_2S_{12} + a_1S_1 \\ 0 \end{pmatrix} \tag{B.7} $$

Finally, the Jacobian matrix of a robot composed of two joints is expressed as follows:

$$ J = \begin{pmatrix} z_0 \times (O_2 - O_0) & z_1 \times (O_2 - O_1) \\ z_0 & z_1 \end{pmatrix} \tag{B.8} $$

From the transformation matrix, all vectors are identifiable.

$$z_0 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = z_1 \tag{B.9}$$

$$O_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{B.10}$$

$$O_1 = \begin{pmatrix} a_1 C_1 \\ a_1 S_1 \\ 0 \end{pmatrix} \tag{B.11}$$

$$O_2 = \begin{pmatrix} a_2 C_{12} + a_1 C_1 \\ a_2 S_{12} + a_1 S_1 \\ 0 \end{pmatrix} \tag{B.12}$$

By replacing these vectors in Eq. (B.8), and performing the vector products[2], the Jacobian matrix is derived.

$$J = \begin{pmatrix} -a_2 S_{12} - a_1 S_1 & -a_2 S_{12} \\ -a_2 C_{12} - a_1 C_1 & -a_2 C_{12} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{pmatrix} \tag{B.13}$$

[2]A vector product is performed in the following way:

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \times \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} u_2 v_3 - u_3 v_2 \\ u_3 v_1 - u_1 v_3 \\ u_1 v_2 - u_2 v_1 \end{pmatrix}$$

Appendix C

PI Controller

A PI (Proportional-Integral) controller achieves zero static error in response to an indexed input and rejects constant disturbances [7]. A PI controller acts via a proportional action, where the error is multiplied by a gain K_p, and via an integral action, where the error is integrated and divided by a gain Γ_i. The command of the PI controller is expressed in the following form:

$$u(t) = K_p \cdot e(t) + K_i \cdot \int_0^t e(x)\,\mathrm{d}x \tag{C.1}$$

where K_p and $K_i = \frac{K_p}{\Gamma_i}$ are the controller gains. The error is the difference between the reference (desired) signal and the measured signal (via the sensor).

$$e(t) = F_{ref} - F \tag{C.2}$$

The block diagram of the PI controller is presented in Fig. C.1 [37].

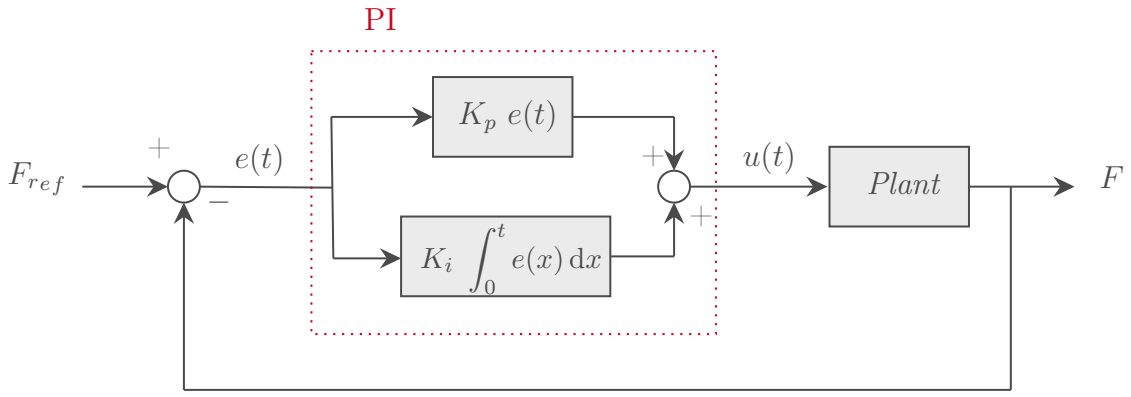


Figure C.1: PI controller block diagram.

Expressed in the Laplace domain, the transfer function of the PI controller is the sum

of both actions.

$$U(s) = \left(K_p + \frac{K_i}{s}\right)(F_{ref} - F) \tag{C.3}$$

A plant is the combination of a process and an actuator. It is denoted by a transfer function, usually in the Laplace domain [35]. The transfer function indicates the relationship between an input signal and the output signal of a system without feedback. In this case, if $G(s)$ is the transfer function of the plant, the following relationship exists:

$$F = U(s) \cdot G(s) \tag{C.4}$$

A transfer function of order 1 is of the following form:

$$G(s) = \frac{\gamma}{\Gamma \cdot s + 1} \tag{C.5}$$

where γ is the static gain and Γ is the time constant to an index input. They are shown in FIG. C.2. The response time is the time taken for the system to settle to its final value when subjected to a step input. It is usually expressed as 95 % of the final value [7].

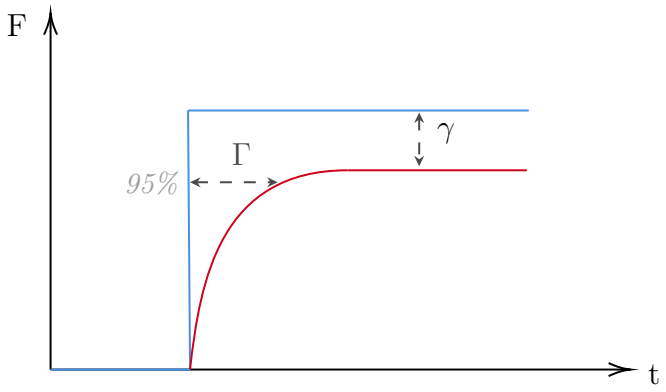


Figure C.2: Static gain and response time to an index input.

By linking the EQS. C.4 and C.5, and performing a change of variable to remove the coefficient in front of s, the following relationship is found.

$$\frac{F}{U(s)} = \frac{\eta}{s + \beta} \tag{C.6}$$

where $\eta = \frac{\gamma}{\Gamma}$ and $\beta = \frac{1}{\Gamma}$. If the EQ. (C.3) is integrated into the EQ. (C.6), a relationship

between F_{ref} and F can be developed.

$$\frac{F}{\left(K_p + \frac{K_i}{s}\right)(F_{ref} - F)} = \frac{\eta}{s+\beta} \tag{C.7}$$

$$\eta\left[\left(K_p + \frac{K_i}{s}\right)(F_{ref} - F)\right] = (s+\beta)\ F \tag{C.8}$$

$$\eta\left(K_p + \frac{K_i}{s}\right)\ F_{ref} = \left[(s+\beta) + \eta\left(K_p + \frac{K_i}{s}\right)\right]\ F \tag{C.9}$$

$$\eta\left(K_p\ s + K_i\right)\ F_{ref} = \left(s^2 + (\beta + \eta K_p)\ s + K_i\right)\ F \tag{C.10}$$

$$\Rightarrow F = \frac{\eta\left(K_p\ s + K_i\right)}{s^2 + (\beta + \eta K_p)\ s + K_i}\ F_{ref} \tag{C.11}$$

The denominator of this function can be compared to that of a second order linear system, $s^2 + 2\xi\omega s + \omega^2$. In order to place the poles of the first and second order function in the same place, their denominators must be equal.

$$s^2 + 2\xi\omega s + \omega^2 = (s+\beta)^2 \tag{C.12}$$

$$= s^2 + 2\beta s + \beta^2 \tag{C.13}$$

$$\Rightarrow \begin{cases} \xi = 1 \\ \omega = \beta \end{cases} \tag{C.14}$$

Finally, the value of the gains K_p and K_i are found by identifying the denominator terms of Eq. (C.11) and Eq. (C.13).

$$K_p = \frac{\beta}{\eta} = \frac{1}{\gamma} \tag{C.15}$$

$$K_i = \beta^2 = \frac{1}{\Gamma^2} \tag{C.16}$$

These relationships allow to find the value of the controller gains when implemented in the closed loop system.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**