UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN BIOINGEGNERIA

"The construction and set up of the exoskeleton for the lower limbs for users with motor disabilities"

Relatore: Dott. Stefano Tortora

Laureando: Riccardo Novello

Correlatore: Ing. Francesco Bettella
Prof. Nicola Petrone

ANNO ACCADEMICO 2023 – 2024

Data di laurea 23/10/2024

Sommario

Le ultime indagini demografiche evidenziano un aumento preoccupante delle persone con gravi problemi di mobilità. Ciò è dovuto alla confluenza di diversi fattori quali l'invec chiamento globale della popolazione, un ricambio generazionale inadeguato, una scarsa prevenzione e stili di vita non salutari. In questo contesto, una tecnologia promettente è rappresentata dagli esoscheletri (EXO) per gli arti inferiori, che consistono in dispositivi robotici indossabili. Questi consentono ai pazienti con grave compromissione motoria di riacquistare la statura eretta o di riabilitare parzialmente le funzioni di base del passo. È stato dimostrato che consentire ai pazienti di mantenere la statura eretta per lunghi periodi senza affaticarsi, riduce l'incidenza di embolie, degenerazioni muscolari, insufficienza renale e disturbi dell'umore.

Durante il seguente progetto di tesi, ho avuto l'opportunità di affrontare le principali difficoltà nella costruzione, assemblaggio e validazione di robot automatizzati. I componenti sono stati progettati utilizzando Solidworks, scegliendo il design più appropriato per garantire prestazioni, stabilità e durata. Elementi quali motori e riduttori sono stati acquistati e montati realizzando 4 giunti motorizzati, rispettivamente due alle anche e due alle ginocchia. Tutti i motori sono dotati di encoder magnetici e sensori di coppia per il controllo. Per completare la costruzione è stato necessario realizzare una serie di elementi aggiuntivi al supporto metallico di base. Tra questi figurano coperture per garantire la sicurezza dell'utente e dispositivi per l'alloggiamento di sensori aggiuntivi.

Successivamente è stato sviluppato il sistema di controllo del movimento per le articolazioni artificiali. È stata utilizzata la piattaforma ROS, Robotic Operative System, per sviluppare ed eseguire tutti i codici necessari. La parte principale del progetto consiste proprio nella messa a punto e nel tuning dei parametri del sistema di controllo. Il controllo si è evoluto da un semplice sistema a controllo di coppia fino ad un controllo completo nella posizione angolare. Tutti i giunti sono stati programmati per eseguire profili angolari ben definiti.

Numerosi test sono stati realizzati per validare il sistema e settare i parametri. Sono state implementate prove al banco, sui singoli giunti in assenza di carico e all'intera struttura. I test sono stati eseguiti su una piattaforma appositamente progettata per sospendere l'EXO.

Prove di carico e coordinazione sono state effettuate muovendo contemporaneamente due giunti per volta, tramite alimentatore da banco. I risultati ottenuti hanno comprovato il funzionamento del dispositivo, la sua efficacia e fisiologicità.

Abstract

The latest demographic surveys show a worrying increase in people with severe mobility problems. This is due to the confluence of several factors such as the global population, inadequate generational change, poor prevention and unhealthy lifestyles. In this context, a promising technology is the Exo-Skeletons (EXO) for the lower limbs, which consist of wearable robotic devices. These allow patients with severe motor impairment to regain upright stature or partially rehabilitate the basic functions of the step. Allowing patients to maintain upright posture for long periods without fatigue has been shown to reduce the incidence of embolism, muscle degeneration, kidney failure and mood disorders.

During the following thesis project, I had the opportunity to address the main difficulties in the construction, assembly and validation of automated robots. Components are designed using Solidworks, choosing the most appropriate design to ensure performance, stability and durability. Elements such as motors and gearboxes were purchased and assembled by making 4 motorized joints, respectively two at the hips and two at the knees. All motors are equipped with magnetic encoders and torque sensors for control. To complete the construction, a number of additional elements had to be added to the basic metal support. These include covers to ensure the safety of the user and devices for housing additional sensors.

The motion control system for artificial joints was developed. The ROS platform, Robotic Operative System, was used to develop and execute all necessary codes. The main part of the project is precisely the development and tuning of the control system parameters. Control has evolved from a simple torque control system to complete control in the angular position. All joints have been programmed to perform well defined angular profiles.

Numerous tests have been carried out to validate the system and set the parameters. Bench tests, individual joints in the absence of load and the whole structure were implemented. The tests were performed on a platform specially designed to suspend the EXO. Load and coordination tests were carried out by moving two joints at a time via a bench-mounted feeder. The results obtained have proved the functioning of the device, its effectiveness and physiologicality.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 State of the Art and Overview

In the field of medical devices for the support or the rehabilitation of people is increasing interest in Exoskeletons (EXO). They are particular devices belonging to the wearable robot class. A wearable robot (WR) is a mechanical device that, employing design and proper functions, can absolve numerous tasks helping the wearer. The definition of exoskeleton is a challenging request because different shapes, functions, materials, and actuators can outline different devices. According to the literature it can be considered an exoskeleton a device externally worn by a human user composed of structural mechanical parts moved through actuators, connected with stripes and cable, that uses electromechanical integration systems to cooperate with the wearers closely. The most evolved one analyzes the human intention through multi-source sensors, solves the motion commands with control strategies designed according to specific requirements, adjusts different forms of actuators to generate suitable power, and finally completes actions in collaboration with human joints [Yongjun Shi and Gao, 2022].

The specific environment defines the tool's goal and functions, which is also true for exoskeletons. The actual force of this sophisticated technology is the possibility to easily convert these devices in shape and functionalities to achieve a new aim and field of work. This is the motor of his rapid increase in industrial environments, special clinique for rehabilitation, inside the military force and private consumers field. In the case of industrial exoskeletons, the primary goals are to augment human capabilities and to prevent weakness or injuries whilst performing repetitive actions [Alberto Plaza, 2020]. In the last decades, the consciousness about work injuries, in particular musculoskeletal disorders (MDS), increased. Numerous strategies, campaigns, and policy initiatives over the last 30 years have aimed to prevent MSDs. However, when MSD prevalence data are examined (collected at the EU level from two Labour Force Surveys (LFSs) conducted in 2007 and 2013), it can be

seen that there was a general increase in prevalence across the 28 Member States of the EU (EU-28), from 54.2%to 60.1%, between 2007 and 2013 [Joanne O Crawford, 2020]. These percentages seem to increase year after year. In the case of Italy, according to the INAIL's report on accidents and occupational diseases, the occurrence of MDS diseases has steadily grown from 64.23% in 2016 to 67.52% in 2020. That encourages the introduction as an operative tool of the exoskeletons inside the industrial factory and environment.

In the same way, numerous studies prove the fact that we are aging, without generational renewal. Work injuries and aging are tightly correlated, steadily increasing the percentage of people who continue to work despite neurological injuries such as stroke, spinal cord injuries, and weaknesses of the skeletal muscles that seriously limit their ability. These conditions affect the elderly and workers, depriving them of completing their daily living activities. A survey from the United Nations shows that people aged 60 and over-represented almost 11.5% of the global population in 2012, and this percentage is projected to nearly double to 22% by 2050, 34% of the European population will be aged 60 and over by 2050. Therefore, along with the worldwide population aging, there is an increasing interest in the use of robotic devices to assist elderly people in accomplishing their main daily living activities as well as rehabilitation exercise training [Weiguang Huo, 2016].

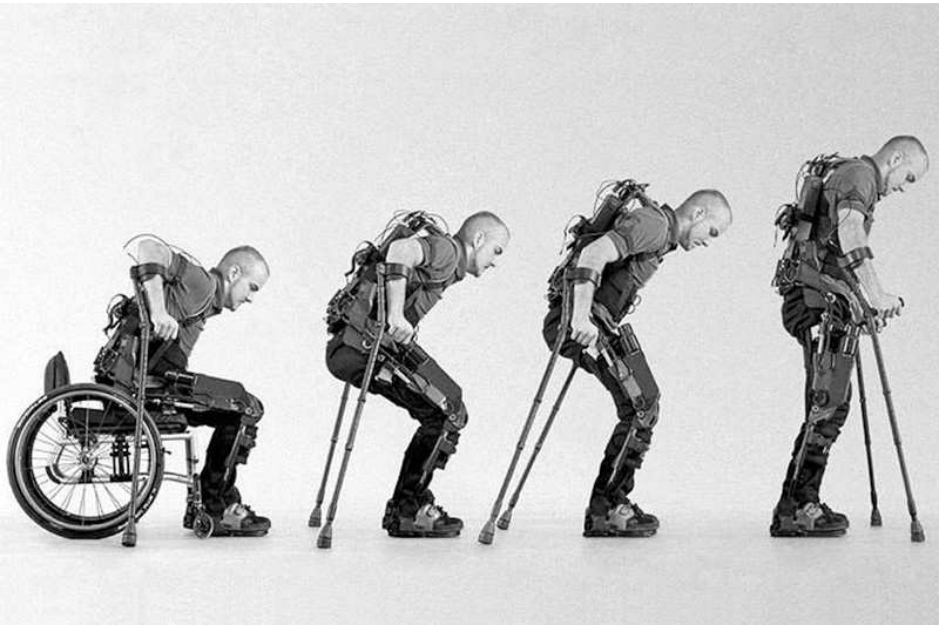


FIGURE 1.1: Paralized man walks with the aid of crutches. [1]

[1] Image Credit: https://nursetimes.org/al-s-raffaele-pisana-di-roma-grazie-allesoscheletro-si-ritorna-a-camminare/2293

The field of assistance and rehabilitation is the most challenging. Exoskeletons, and orthoses, have difficulties interfacing with people who suffer from paralysis, lack of a limb, and limited range of motion. Usually, patients who suffer from these diseases are not very collaborative with the rehabilitation robots and with exoskeletons; they feel uncomfortable and, in most cases, they have pain in correspondence of joints and linking. But, it's a matter of fact that, assistive robotic exoskeletons can allow users to complete movements they could not complete on their own. For example, many of these exoskeletons are intended to enable an individual with lower limb paralysis to walk with the aid of crutches [Figure 1.1]. Moreover, these devices can assist, resist, or perturb the user's movements to achieve therapeutic exercise. They can train an individual's muscles and/or nervous system to help them overcome the limitations of a disability when they are not using the exoskeleton [Young and Ferris, 2017].

During the pandemic, it was found that there was a need for some systems like EXO that could help people when the operator or a loved one cannot intervene. This was a new borne force to the research for the assistant EXOs for indoor and outdoor envinments. These systems aim to complete the motions and intentions of the user during the everyday routine, encouraging users' independence and re-founding the serenity inside their home. The use of these devices has a significant advantage for users among the most important we find:

- Regain the upright stature;
- Reduces the incidence of embolism;
- Arrest degeneration of muscles;
- Reduce kidney failure;
- Rehabilitation of simple motion.

In this context, a promising technology is represented by exoskeletons (EXO) for the lower limbs which consist of wearable robotic devices, they represent the future of the standard treatments and assistant robots. The EXO market has grown strongly in recent years, with a global market projected to grow from $1.24 billion in 2023 to $14.67 billion by 2030, at a CAGR of 42.2% [2]. Despite recent developments, this technology is relegated exclusively to clinical use for rehabilitation purposes. The difficulty of transferring exoskeletons from rehabilitation tools to assistive robots is due to the following well-known limitations:

[2]Data from: https://www.fortunebusinessinsights.com/wearable-robotic-exoskeleton-market-10466

1. The request for a passive support system (e.g. crutches, rollators) to allow balance and rotation during walking. State-of-the-art exoskeletons have no or extremely limited self-balancing capabilities, thus making their use unsafe in uncontrolled environments and the absence of a physical therapist.

2. The inability of the EXO to perceive the environment in which the movement takes place and to provide active support during rotations, leaving entirely to the user the task of navigation (e.g., avoiding obstacles), increasing his workload, particularly in real environments.

3. Poor investigation of shared control systems in wearable robotics, in which the robotic device is seen only as a mere actuator of a user command.

To improve the style and standard of living of many people, it is clear that the commitment to the development of these systems must be increased. Our job as researchers is to engage the actual problems and discover a new way to exploit this emerging technology.

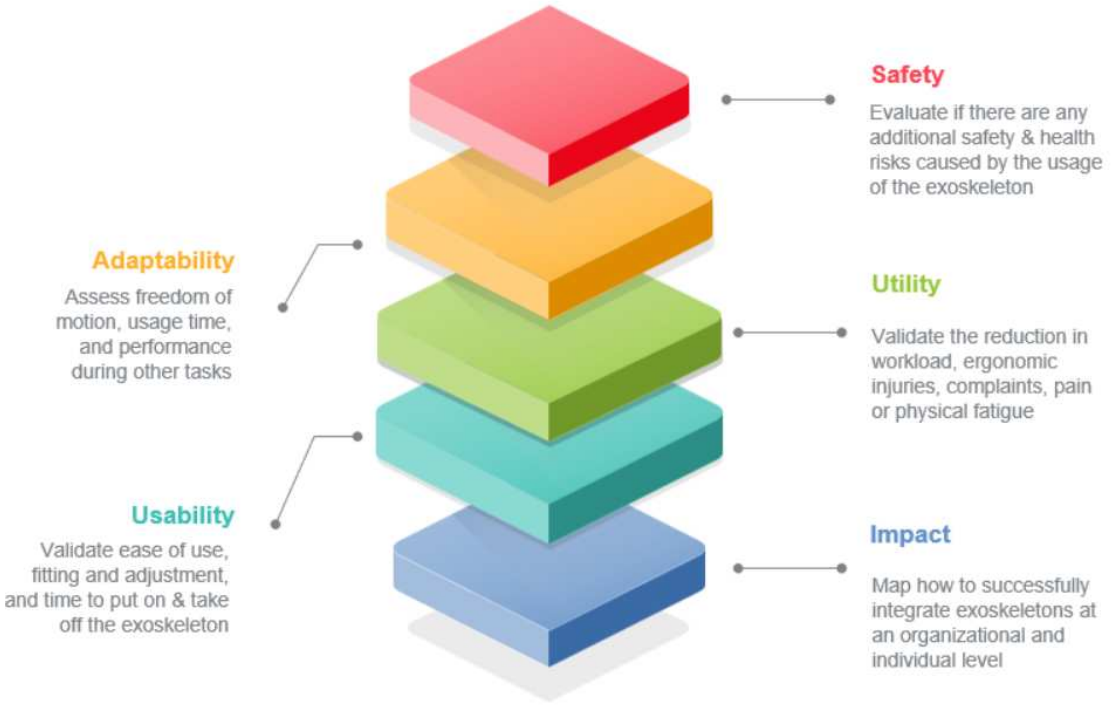


FIGURE 1.2: Main elements within a holistic exoskeleton evaluation.[3]

[3]Image Credit: The Ultimate Guide to Successful Exoskeleton Implementation

The research is fully applied to the development of a device powerful and attractive for the user, useful even in the most difficult environments, able to adapt to unforeseen contexts, and that it is always safe for those who use it or for those who are nearby.

1.2 Biomechanics of human locomotion

When it comes to adaptation, the human body is the most capable system. The structures of tissues, bones, and joints give our body the strength and the agility to overcome obstacles, and different grounds and to make a large number of actions. When it comes to lower limbs, many features distinguish them and make them unique. The lower limbs can bear body weight for a long time, they prevent the burnout of the spinal cord, make us free to move easily and rapidly, and give us the possibility to achieve complex control when two hands aren't enough. The anatomy and physiognomy of lower limbs are ideal to guarantee stability and equilibrium also during movement or with one leg. It is commonly known that they are very important also for the health of the human mind, from the psychological point of view they represent the ideal of liberty and to be free, the ability to discover the world step-by-step. Unfortunately, it is also clear that as sophisticated and formidable are these human native tools, as fragile and difficult to heal they are.

Surgery and pharmacological treatments can cure a good part of diseases, but unfortunately, in the case of strong trauma, paralysis, stroke, and neurodegenerative diseases, the actions remain an unresolved field. Currently, neurodegenerative diseases, such as Alzheimer's Disease (AD), Multiple Sclerosis (MS), Parkinson's disease (PD), Huntington's disease (HD), dementia, etc., are not curable. The World Health Organization (WHO) predicted that by 2030, neurological disorders will represent the second leading cause of death, worldwide. Currently, available treatments can only limit the rapid progression of the disease [Grazia Cicirelli, ANUARY 2022]. In the best-case scenario the patient, who was cured, can restore only a few movements or can partially recover his daily routine. Is understandable that in these cases is necessary to include the patient in a rehabilitation protocol or to give him specific tools for his assistance.

In this field, the most important condition, that's the main condition of exercise, is the walk. Restoring the walking gait is the final goal for reintroducing the patient into society as an active person. Robotic gait rehabilitation appeared 25 years ago as an alternative to conventional manual gait training. The advantages brought by wearable exoskeletons are extremely useful to the therapy for gait rehabilitation, compared with conventional treatments

from physiotherapists, they can deliver highly controlled, repetitive, and intensive training in an engaging environment, reduce the physical burden for the therapist, and provide objective and quantitative assessments of the patient's progression. Moreover, the capability of these robotic devices to draw attention from the patients involves them in active participation and promotes physical activity and the possibility of being used as an assistive device in the community [Antonio Rodríguez-Fernández and Font-Llagunes, 2021]. Unfortunately, the actual statistics of successful rehabilitation at the end of the treatments are very low, only a few parts of all the patients can restore the lost neuro-muscular abilities. In this case, the alternatives are represented by the assistant EXOs [Figure 1.3].

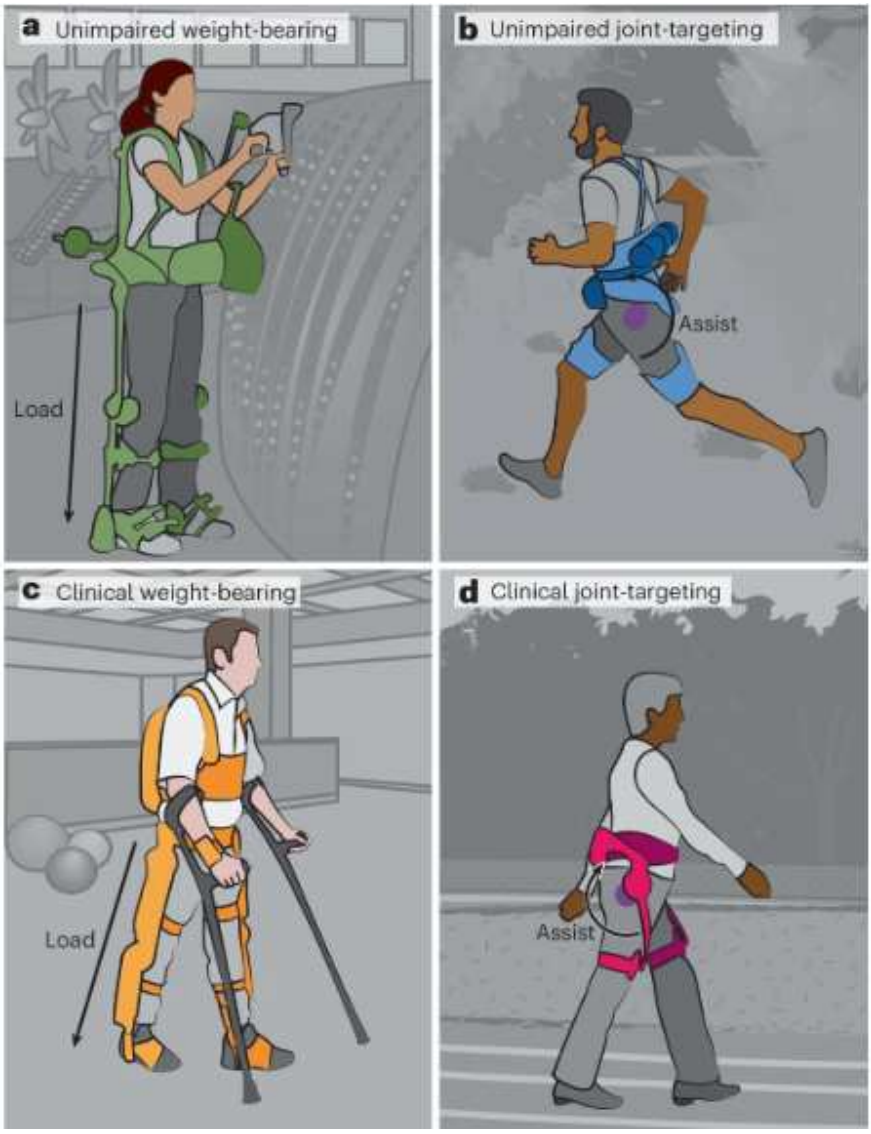


FIGURE 1.3: Examples of the assistive exoskeletons during use of daily life [Christopher Siviy, 2022].

Assistant exoskeletons are very similar to the industrial ones in the aim but dissimilar in complexity. The principles are the same as the rehabilitation EXO but in this case, the device must have more characteristics to support the patient and also to coexist with it without injuries or contrast during normal daily activities. The force of these technologies is the ability to emulate physiological movements, to collaborate with the patient looking for the autologous path, and to sustain him during the actions. But, these features are at the same time the limitations of these devices. Because, the proper functioning of the EXO involves synchronizing the human joints with the mechanical actuators, fixing all the structures of the specific body part with tapes, and establishing a defined path for control. Is a matter of fact that a lot of users complain of pain in the joints, and difficulties in movements, that's because there are limits to technical implementations, like not the proper design of the shape, functionality of motor, or disengagement of joins.

To prevent all of these drawbacks, is necessary to better understand human physiology and anatomy to achieve a proper design:

- Study the mechanism of native joints to finally establish an actuator that has the same degrees of freedom.
- Define an interface human-machine that can overcome the mutual impediment to the natural walking process.

To solve all these problems are trying to use innovative artificial intelligence (IA), with a soft algorithm that learns from you the typical expression of movements and necessities of your body. With that, it's possible to approach a configuration of patient-specific control that avoids the mutual contrast. Moreover, is necessary to take into account a lot of conditions that are usually relegated to second place, and that are strictly correlated with anatomy, like synergies of more muscles, reflex mechanisms, muscle and/or cutaneous feedback, and balance during walking on uneven surfaces [J.L. Pons, 2013].

1.2.1 Anatomy call back

The first step is to define the references to the human body, like a vector triad inside a 3D space. Medical institutes commonly use anatomical planes as a standard. An anatomical plane is a plane for dissecting the body to describe the position of structures or the direction of movements. In human anatomy, there are three main planes for the correct and coherent description of the body [Figure 1.4]. They are the following:

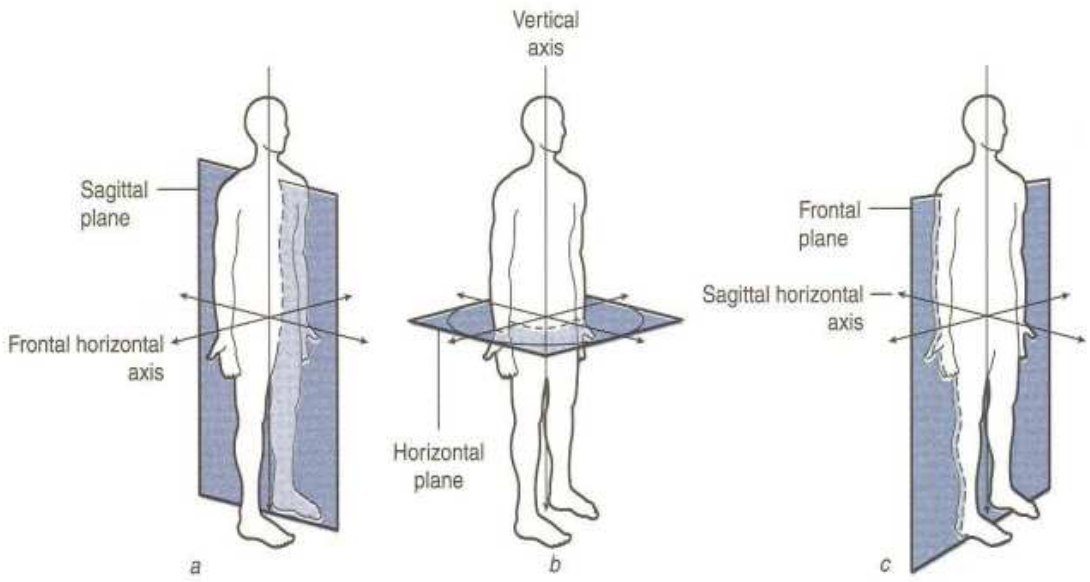


FIGURE 1.4: Representation of human body part with plane and axis intersection.[4]

- The sagittal plane (longitudinal, anteroposterior), a plane parallel to the sagittal suture. It divides the body into a left and a right part.
- The coronal plane or frontal plane (vertical) divides the body into a dorsal and a ventral portion (anterior and posterior respectively).
- The transverse (horizontal) plane divides the body into a cranial portion and a caudal (upper and lower respectively).

Due to the previous definitions, it's possible to enhance the specific reference with the axes. An axis is a straight line around which an object rotates. Movement at a joint takes place in a plane about an axis. Like the planes, there are three main axes to define [Figure 1.4]. They are the following:

- Sagittal axis, it passes horizontally from posterior to anterior and is formed by the intersection of the sagittal and transverse planes.
- The Frontal axis, passes horizontally from left to right and is formed by the intersection of the frontal and transverse planes.
- Vertical axis, it passes vertically from inferior to superior and is formed by the intersection of the sagittal and frontal planes.

[4]Image Credit: https://thalessyalves.com.br/index.php/2024/03/19/plano-e-posicoes-anatomicos/

All of these are essential to describe movements, define paths, and control output. Every single step can be divided into projections inside the three planes and can be established as a mutual reference with the axis. Talking about walking, this action is commonly considered a single movement on the sagittal plane, but in reality, this is only a description of the gross direction of movement. Movement will take place in all three planes at the individual joint level, not just in the sagittal plane. During walking, the hip will be flexing/extending in the sagittal plane, adducting/abducting in the frontal plane, and internally/externally rotating in the transverse plane.

If we take a look at the hip, it's almost obvious that during the gait, the main plane that contains the movement is the sagittal plane. But the hip joint makes three types of movement during walking and the forces that occur during the action might be expressed in all the three planes. That forces play like antagonist scheme, is an interplay between the eccentric force absorption and the concentric force. Internal rotation and adduction are being decelerated by the hip, while external rotation and abduction are being accelerated. This simultaneous movement can be seen as one motion with three components, it can be termed tri-planar motion [5].

Plane	Motion	Axis	Example
Sagital	Flexion/extensionn	Frontal	Walking Squatting Overhead press
Frontal	Abduction/abduction Side flexion Inversion/eversion	Sagital	Star jump Lateral arm raise Side bending
Transverse	Int rotation or ext rotation Horizontal flexion/extension Supination/pronation	Vertical	Throwing Baseball swing Golf swing

TABLE 1.1: Examples of dominant planes, motions, and axis in gross movements.

This opens another deepening, of the types of movements that we can realize with our limbs. These are extremely connected to the type of joint and their degrees of freedom (DOF). The classification of joints by healthcare providers is based on the amount and type of connective tissue they possess. Three general types of joints:

- Fibrous joints.

[5]Data from: www.physical-solutions.co.uk

- Cartilaginous joints.
- Synovial joints.

They are distinguished by the strength of components, concentration of collagen, resilience, cushioning, and rigidity. But for the current analysis, the joints that operate in the walking gait and generally for the ambulation, are included in the last ones of the list.

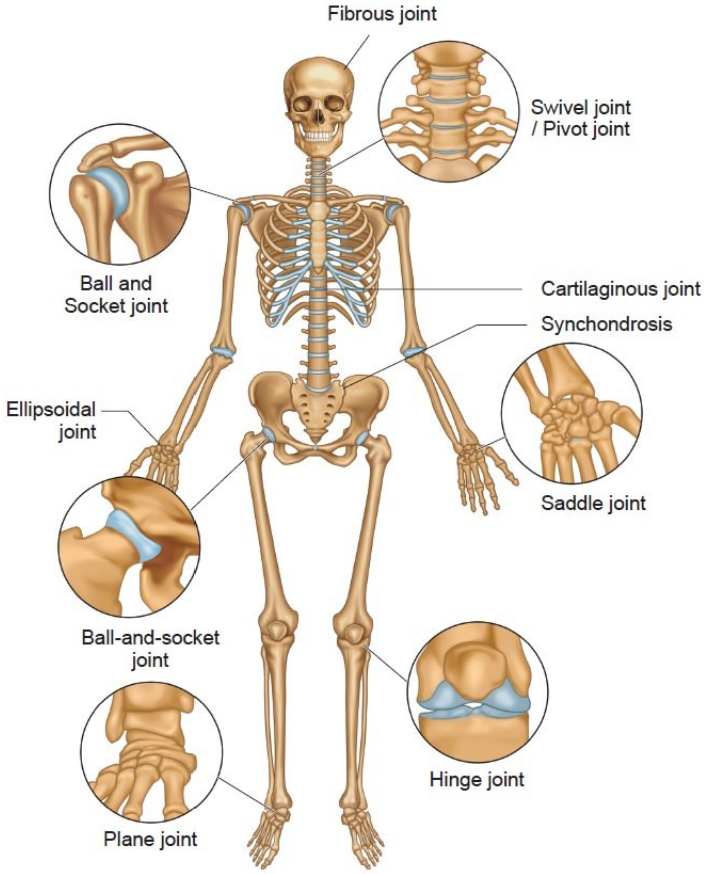


FIGURE 1.5: Structure and Types of Joints in the Human Body. [6]

Synovial joints have the most freedom to move. The base of the mechanism is the synovial capsule. The synovial capsule consists of a synovial membrane (a fluid-filled sac that lubricates and protects the joint), and the bones, that are not directly in contact but covered, in the end, by slippery hyaline cartilage. This condition guarantees free slip during the movements, synovial joints move with as little friction as possible [Juneja P, 2024]. There are six types of synovial joints:

[6] Image Credit: $https://www.brainkart.com/article/Types-of-joints_{3}3251/$)

- Hinge joints: Joints that open and close in one direction. Your knees and elbows are hinge joints.
- Ball and socket joints: In a ball and socket joint, the rounded end of one bone fits into an indentation in another bone. They can rotate and turn in almost any direction. Your shoulders and hips are ball and socket joints.
- Condyloid joints: Condyloid joints are made of two oval-shaped bones that fit together. They're similar to ball and socket joints, except they can't rotate in a full circle (360 degrees). Your wrist and the joints where your toes meet the rest of your foot are condyloid joints.
- Pivot joints: Pivot joints rotate in place without moving out of their original position. A pivot joint in your neck lets your head move from one side to another.
- Planar joints: Planar joints are formed when two mostly flat bones come together. They move by one piece of bone sliding over the other without rotating. The carpal bones that join your wrist to your forearm and the joints between the vertebrae in your spine are planar joints.
- Saddle joints: Saddle joints are formed when two curved bones meet. Picture two U-shaped bones fitting together in the curved space between each other. Saddle joints can move in any direction, but can't twist or rotate. The joint where your thumb joins your hand is a saddle joint.

Ball-and-socket joint, Plane joint, and Hinge's joint are the functional joints for ambulation, standing, and running. Ankle and knee joints are the simplest to emulate with an actuator inside the EXO because the movements and the DOFs are similar to the function of a one-axis motor. In general, the human leg configuration consists of seven degrees of freedom (DOF) structure in each leg, three rotational DOFs at the hip, one at the knee, and three at the ankle, but is also real that some studies have proved the bilateral model presented 18 DOF in total [Bhaben Kalita, 2020]. The choice of the reference depends on the degree of complexity with which you want to operate.

But in the case of hips, we usually do an approximation. We consider only two types of possible movements at the hip level, extension and flexion; sometimes the researcher includes hyper-extension, but this condition is acceptable only with non-pathological subjects. This type of approximation is essential to build a solid structure without lability improving equilibrium. From the anatomical point of view, it's a strong constraint, that in a lot of cases can create difficulties and pain. Also for the knee and ankle, we usually do approximations.

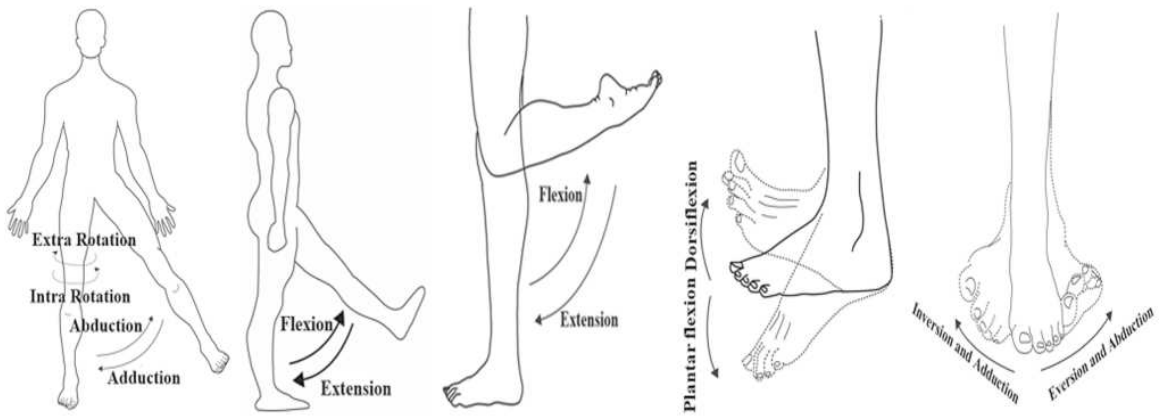


FIGURE 1.6: Diferent joint movement of the lower limb [Juneja P, 2024].

But in the case of the ankle, we have dorsiflexion and plantar flexion, abduction, and adduction, we reduce a lot of movements. However, in this situation, it's not considered an extreme simplification, because these types of labilities aren't essential during a normal gait of walking on a flat ground. They occur when the ground is arduous and the stability of the subject is affected.

Joint	DOFs	Range(°)
Hip	Flexion/extensionn Abduction/abduction	-120/+65 (-30/-35)/(+40)
Knee	Intorsion/extorsion Flexion/extension Adduction/abduction	(-15/-30)/+(60) (-120/-160)/(0) (-30/-35)/(+15/+20)
Ankle	Intorsion/extorsion Dorsiflexion/ plantarflexion	(-15)/(+30/+50) (-20)/(+40/+50)

TABLE 1.2: The range of DOFs of each joint.

Depending on the needs of users, can be build an EXO full of these features, that can adapt to the different contexts and make various actions when necessary. The environment and the aim establish the effective design and the ability of EXO. Of course, the active motion of the EXO must be coherent with the range of motion of the human joints. The motion range of each joint of the lower limb exoskeleton robot should match that of the human body's lower limb joints. These conditions correspond to the DOFs of the human limbs shown in Table 1.2 [HE Guisong, 2022].

1.2.2 Gait Analysis

During a walk, the rapid sequence of lower limb movements, along a forward way direction, is commonly called the gait cycle. The motion is made by the combination of movements of the anatomical districts and joints. Formally, the gait cycle is defined as the interval between two successive heel strikes of the same foot (step), and is possible to define it at any moment inside the walk activity, but usually, the start condition is correlated with the foot. The starting point and the endpoint are established with the contact of the heel with the ground of the selected foot [HE Guisong, 2022][Bhaben Kalita, 2020]. As shown in Figure 1.7, the cycle is composed of two macro sequences:

- Stance phase. In general, takes around 60% of the gait cycle and can be divided into double support and single support. In double support, both feet are in contact with the ground, while in single support only one foot is in contact with the ground. The stance phase includes the heel-to-toe contact sequence of the foot.
- Swing phase. It's described when the limb is not weight-bearing and represents around 40% of a single gait cycle. The swing phase refers to the movement of one leg off the ground while the human body is walking.

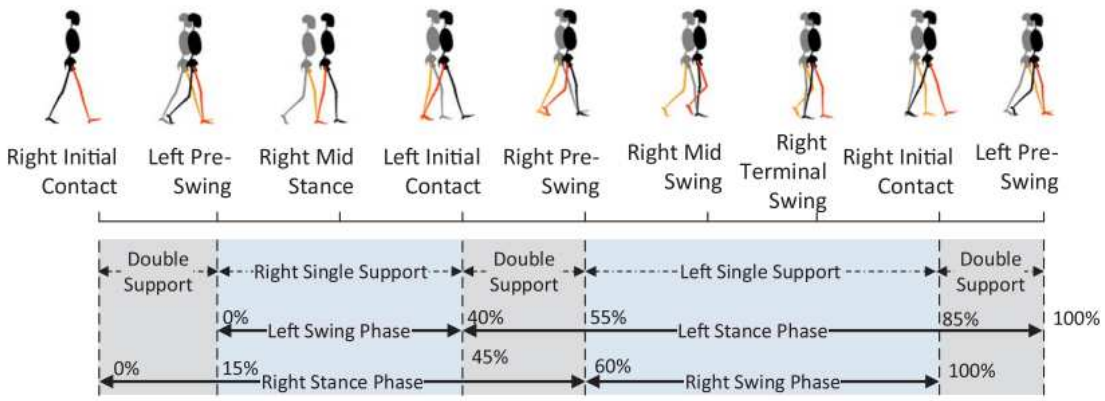


FIGURE 1.7: Gait pattern of normal human walking for one cycle [Ioannis Papavasileioua, 2017].

Double or single support ratio refers to the portion of time within a gait cycle someone spends in double or single support respectively. The percentages are calculated from statistics means of the sample population, they can be different with the walking speed, pathology, or with the presence of limb diseases. As is shown in Figure 1.8, the main phases can be divided into 8 micro sequences for a healthy subject: initial contact, loading response (or pre-swing),

mid-stance, terminal stance (or initial contact), pre-swing, initial swing (not shown), mid-swing, and terminal swing. Both feet are in the supporting phase simultaneously, accounting for about 10%. This type of succession of phases can be lost if we consider pathological gait, the time allocation of gait phases might also be different from a normal gait [Ioannis Papavasileioua, 2017].

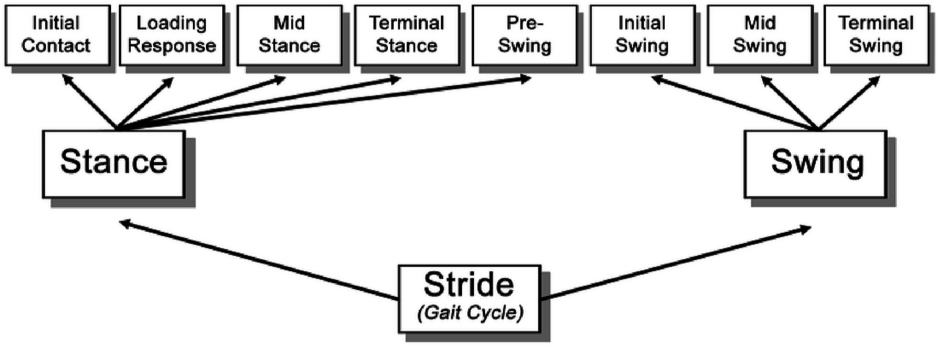


FIGURE 1.8: Main phases divided into 8 micro sequences for a healthy subject [Blaya, 2005].

From the gait cycle, is possible to know some spatial parameters, correlated to the process of human walking: step length, stride width, step width, and step frequency.

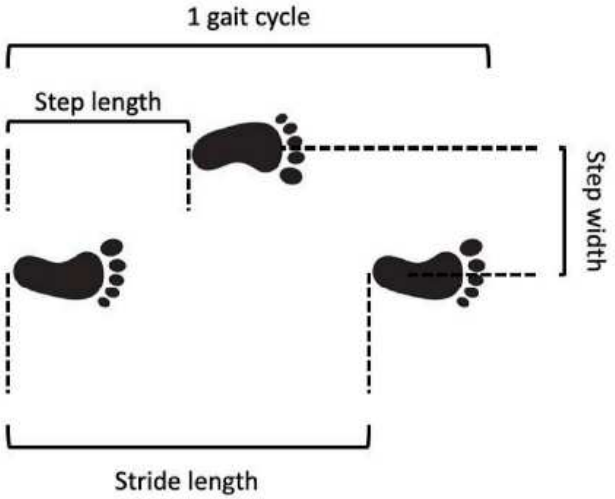


FIGURE 1.9: Gait cycle anthropometric measure of step[Alessio Martinelli, 2015].

Generally speaking, the step length of a person who is walking is between 500 mm and 800 mm, the straight-line distance between the heels of two feet can change with height or velocity. At the same time, the stride width was proven to be one and a half or twice the step

length, it refers to the straight-line distance between the heels of the same foot; step width refers to the lateral distance between two feet, generally about 50—100mm; step frequency refers to the number of steps taken per unit time during human walking [HE Guisong, 2022]. As shown in Figure 1.9, step frequency can be represented by a succession of three steps, two for the right leg, one fore left [Tanner Amundsen, 2022].

To effectively detect the gait phase and collateral information, is essential to include in the specific analysis different types of sensors and algorithms. It's necessary to receive information from sensors to develop a clear signal of movements and classify all the gait phases. In the literature are well known a lot of sensors that have been used for real-time data acquisition. They can be classified into two main categories:

- Wearable Sensors, are the most common, they can be placed on different parts of the patient's body, like joints, muscles, and ends; the captured data are of the local space of the body part. Data are usually transmitted through wireless connections or collected onboard storage devices. The main wearable sensors used for gait analysis are wearable inertial sensors like accelerometers, gyroscopes, and magnetometers.
- Ambient Sensors, they are decoupled from the body, but mounted in the environment. The most commonly used for gait analysis are force sensors, pressure sensors, and vision-based sensors.

The force and pressure sensors are very useful is possible to measuring the Ground Reaction Forces (GRF). It's associated with walking and the Center of Pressure (CoP) of the body, and is acquired by the sensoring platforms, or instrumented walkways. The GRF and the CoP are important elements to define the motion gait and also to measure the direction and types of the gait in the three-dimensional space. Usually, to achieve a more complete analysis of gait, is necessary to include another known system, the Motion Capture System (MCS). Extending the analysis with both types of information, and converging on a single solution can enrich the model [Grazia Cicirelli, ANUARY 2022].

1.2.3 Kinematics Analysis

To obtain the real condition of the assisted motion function, we need to include in our analysis other objects that vary from the simple DOFs. DOF analysis can only meet the anthropomorphic design, it's necessary to build a mathematical model that achieves the coordination of the exoskeleton robot and the lower limbs of the human body. The exoskeleton robot's movement posture is determined by the model's description of the position relations of each

joint in the space coordinate system, which is necessary for kinematic analysis.

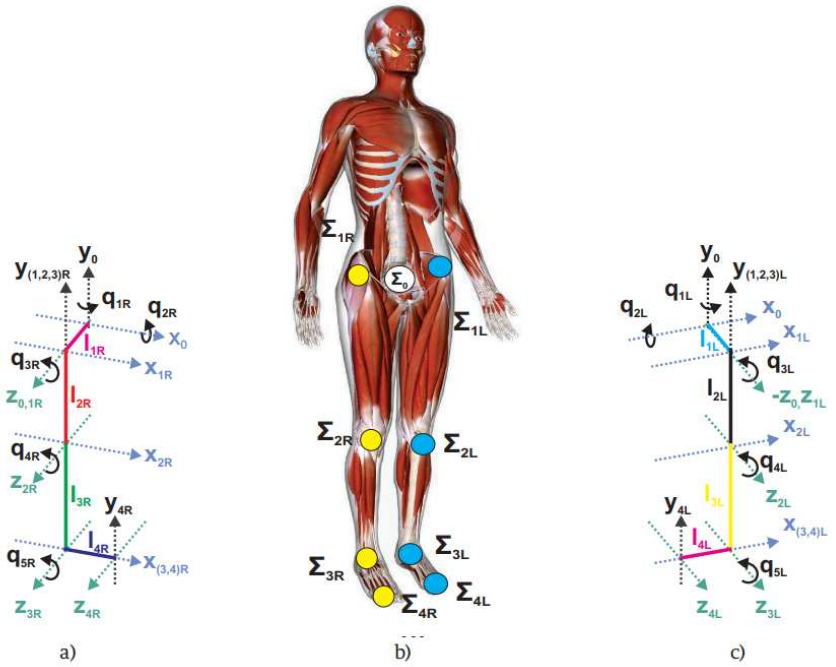


FIGURE 1.10: Analysis method of Denavit-Hartenberg, axis, and terns draw.

A simple method to study mechanical kinematism made of rigid pieces joined each other with continuity of function, that exhibits specific degrees of freedom and movements, is the coordinate analysis method of Denavit-Hartenberg (D-H) [Figure 1.10]. It is commonly used to study and create robots of industrial environments like SKARA or anthropomorphic arms to help workers, but it's also used to design clinical robots like the DaVinci or the CyberKnife. In this case is possible to study and simplify the lower limb of humans with this method, obtaining the data to establish a mathematical model of human-machine coupling. Therefore, this model can perform the kinematics analysis on the exoskeleton robot, and obtain the relationship between the exoskeleton robot-end and each joint motion parameter [HE Guisong, 2022].

Procedure:

1. The axes of movement, related to rotation or translation of the kinematic couples, shall be identified and numbered. In the drawing you have to trace them, the first axis is that which binds the first member to the frame, in this case, the upper body;

2. The normal common to the pairs of successive axles shall be identified, in the case of bidirectional axes, the direction is obtained with the carrier product and passes through two points defining the minimum distance between the axes;

3. Draw the reference terns (first the origin, then the z-axis and the x-axis, then the y-axis according to the right-hand rule);

4. Last, a table (the DH table) is completed containing in each row the four parameters of a relative transformation; the number of rows will be equal to the number of degrees of freedom of the manipulator; each row will contain one and only one free coordinate (one angle if the joint is rotating, one distance if it is translational) and three constants.

Despite of popularity of the Denavit-Hartenberg convention and geometric methods, which are extremely useful to treat robots, when the subject is the human body these models are not able to describe correctly the kinematics and the gait, it has been said that "these types of methods wast quaternions capacity". Is a matter of fact that, the Denavit-Hartenberg convention and geometric methods can solve well the forward kinematics of position in the sagittal plane of the 3 DOF joint system of the lower limbs. But, it is real that, in case of an increase of DOF, like in the condition of 8 DOF, or taking into account different view planes, the complexity of the model starts to be a high computational burden and loses the flexibility in the description of the evolution of the system of the extremities. A possible solution comes from quaternion algebra as a mathematical tool.

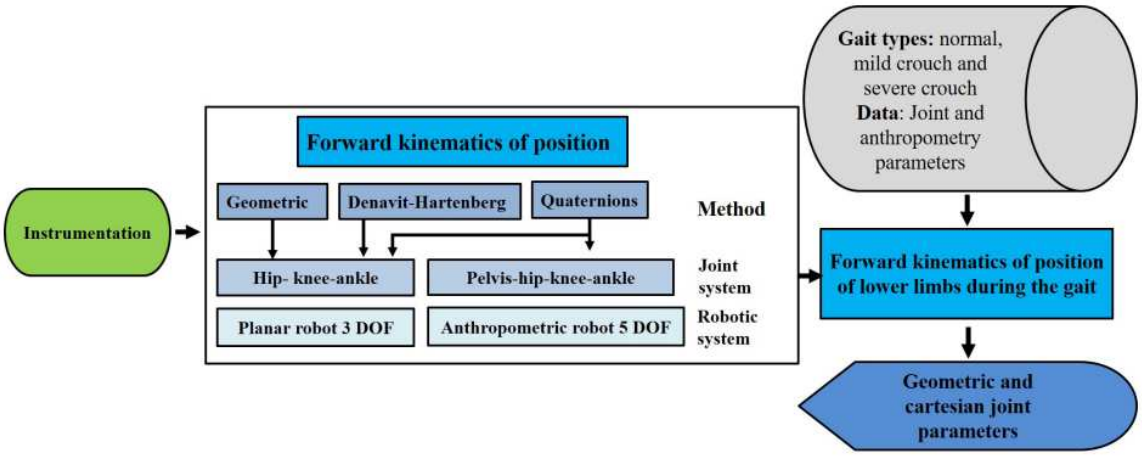


FIGURE 1.11: Methodology for the modeling and visualization of the forward kinematics of position of the lower limbs during gait [J. C. González-Islas, 2020].

Combining the precedent methods and joint sensors data acquiring with quaternions is possible to increase the complexity of models and to define an anthropometrically coherent solution [J. C. González-Islas, 2020]. The development of quaternions for the forward kinematics 8 DOF system is due to the increased complexity (two legs and pelvis joint), and a simulation is conducted to visualize the cartesian coordinates and joint variables of both extremities [Figure 1.11].

This type of model is the base to build an EXO for the lower limbs. Finding the principal axis, the nodes of torque, the vectorial tern of the joint, and the matrix of translation/rotation are the main subjects of designing an EXO. But, the kinematics analysis covers the first part of the total design's aspects. To reproduce a movement with the joint we have to know the entity of the movement and the force acting during the action. The conjugation of the two aspects, kinematics, and dynamics, can be done only with the declaration of the specific gait, so is necessary to conduct gait planning to establish the exoskeleton robot's specific gait trajectory.

1.2.4 Acquisition gait reference

The most common strategy to study human biomechanics, and acquire information on gait cycle and dynamics relationships, is to use a Motion Capture System (MCS) combined with power platforms. The curves that plot the angular range of motion on the support period time during the walking are the specific input to control the EXO's movements. In this field, there are different ways to analyze the human walking cycle, however, only two specific procedures and tools are powerful enough to solve this task:

- Marker-based (MB) motion-capture system.
- Markerless (ML) motion-capture system.

The motion-capture marker-based system uses the inverse kinematic and kinetic data with anthropometric parameters to estimate joint moments and powers, angular excursion, and member's gait. The system is usually composed of several camera 3D (stereophotogrammetric cameras) and a 3D force plate for the ground reaction forces. It's the most used strategy of the last decades and is considered a gold standard for human kinematics analysis. Although it is widely used, reverse dynamic analysis that is based on kinematic/kinetic/anthropometric data is well-known for its inaccuracy. This type of tool suffers from some disadvantages, like:

1. Noise due to surface marker movement

2. Skin artifacts
3. Marker placements
4. Center of mass locations
5. Joint centers

Most of these are limits due to the actual technology systems, also the most advanced motion-capture setup suffers from them, and that results in a dead end. But is a fact that this tool is the most evolved and accurate one, unless his rival.

As an evolution in vision-based motion capture, a marker-free (ML) system was born, offering an alternative approach to kinematic data measurement. Studies have shown the applicability of the markerless system based on deep learning algorithms in gait analysis. This system uses computer intelligence to understand and predict the gait cycle. Currently, is a newborn technology that is not widely used, but is clear that has all the conditions to overcome the MB limits. Studies showed how the intersession variability of joint angles was very low for the ML approach concerning the MB. Unfortunately, the computational burden, specialized personnel, and memory space required are too high/expensive for some research groups or laboratories, so this technology is still a niche tool.

With this introduction and disclaimer, I motivate the choice to use as examples reference angular motion curves from a motion-capture markerless-based system, using a dataset from a professional analysis of the human biomechanical gait cycle during normal walking. The importance of these analyses is the possibility of creating a mechanical design and software control that can be personalized when necessary. The actual medical science aims to achieve the patient's specific cure or dedicated assistance.

Robert M. Kanko et all, have studied a court of 8 people 6 male and 2 female with a markerless-based system during the walking. They used a deep learning algorithm-based approach and deep convolutional neural networks for object recognition (humans and human segments) within 2D camera views. For a given image, the network assigned high probabilities to labeled anatomical feature locations and low probabilities elsewhere. The images are acquired from a video recorder during the total gait cycle of the subject [Robert M. Kanko, 2021].

From the collected videos, Theia3D, the name of the system, obtains the 2D positions of its learned features in all frames of all videos and then converts them to 3D space based on

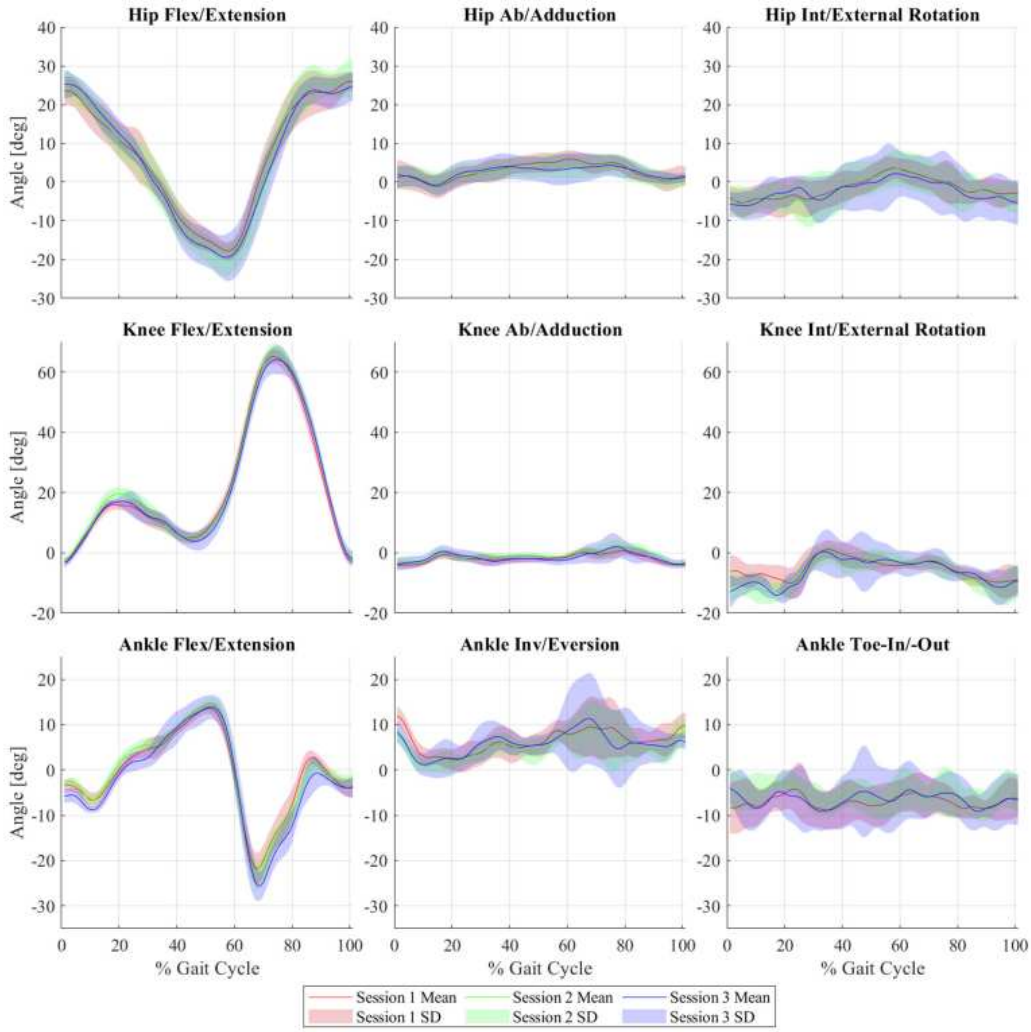


FIGURE 1.12: Motion-capture markerless-based system human biomechanical gait cycle during normal walking, focus on the knee, ankle, and hip [Robert M. Kanko, 2021].

the computed position and orientation of the cameras. Lower limb joint angles were calculated using the standard Cardan rotation sequence (X-Y-Z), equivalent to the joint coordinate system, and time-normalized to the gait cycle using the heel-strike events. Walking cycles were analyzed for both the right and left gait, within a 5-meter long path. The scan angles are common to all analyses and normalized times. For each test and each joint, the data analyzed in MATLAB were exported. Were done multiple sessions for all the cycles [Figure 1.12].

The results show the solidity of this system and method. They have demonstrated how the new technological approach can achieve complex tasks with a tiny discrepancy referring to the actual gold standard.

- Hip: The angle of the hip joint is defined as the angle between the pelvis and the

	Mean value (SD)
Age:	30.3 (14.1) years
Height	173.8 (9.0) cm
Mass	69.0 (12.4) kg

TABLE 1.3: People's data to establish antrophometric measure and configure the acquisition [Robert M. Kanko, 2021].

segment of the thigh. The hip movement starts at the heel and ends at the toe. The plane where the movement is greatest is the sagittal plane with flexion and extension. The maximum hip extension occurs immediately after the change of the podalic support. This is due to the combination of the movement of the pelvis and the angle that develops with the thigh.

- Knee: The main movement is also in this case in the sagittal plane involving the bending and extension motion. In the sagittal plane, the extension and flexion of the knee are repeated cyclically. The range of movement varies from 0 to 70 degrees, these values are indicative depending on several factors such as speed of travel or more generally by the subject.

- Ankle: The most significant plane in which movement occurs is mainly the sagittal plane although in this case movements in other planes also contribute. The movement in the sagittal plane is fundamental as it allows the absorption of the shocks to the heel during the phase of positioning and subsequently allows the propulsive force before the toe leaves the ground.

Moreover, the model is capable of catching user movements and building curves patient-specific. The curves are fundamental to realizing a fully functional EXO for lower limbs. The input for the joint is the curves for Hip Flex/Extension and Knee Flex/Extension. The role of the programmer is to create a database of all these curves and apply machine-learning algorithms to learn the system what to move, when, and how. A full understanding of these movements' profiles is essential.

1.2.5 Dynamics reference

As said before, the analysis of the dynamic is a fundamental part to understand forces that are in game during the gait cycle. The torque and the current delivered to the EXO are strongly influenced by the inertia and the reaction forces presented during the cycle. Goals of these analyses are to obtain the force's profile. With that, the programmer can implement the right balance control.

As references, I chose to take into account the study of the ground reaction forces (GRF) recorded during the stance phase of two steps, obtaining the magnitudes of reaction forces acting on the human body [František Vaverka, 2015]. Vaverka et all used a sensing force plate to map the force profile during the gait and to decompose this force into three components:

- Fx – mediolateral (ML);
- Fy – anteroposterior (AP);
- Fz – vertical.

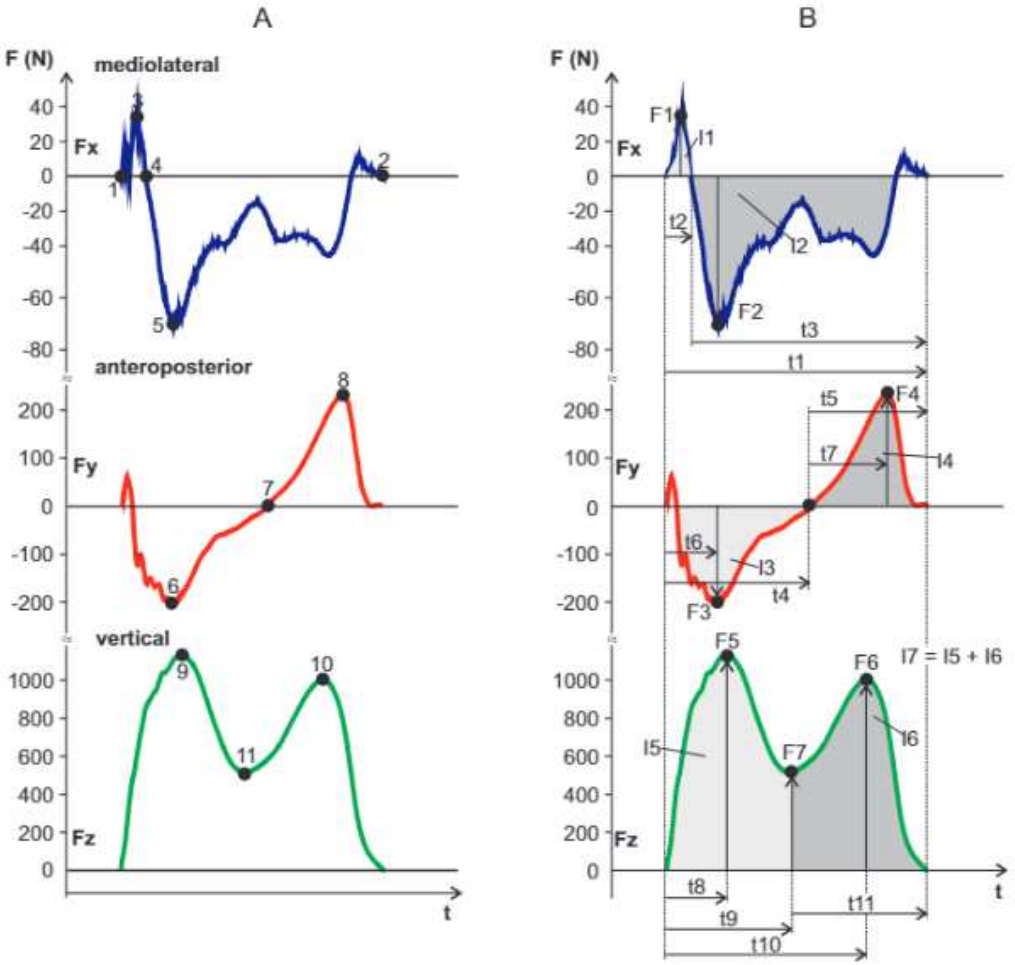


FIGURE 1.13: Ground reaction force during the stance phase of one step (A). Graphical illustration of temporal and force variables of the three GRF components (B), [František Vaverka, 2015].

The ML component determines the balance in the mediolateral plane, the AP component determines the deceleration and acceleration of the movement, and the vertical component,

which acts perpendicular to the ground, is antagonistic with the gravitational force. The method used quantifies during the two-step stance phase the time variables of the function extremes Fx (t), Fy (t), Fz (t), the duration of the braking and propulsion phases, the double support, and the total duration.

In a standing position with both feet on the ground, the reaction force of the soil will be halved in the two supports and equal intensity with equal supports. Whereas in conditions where there is an imbalance, for example lifting a foot off the ground, the reaction of the soil will be asymmetric. During walking there is a continuous and gradual exchange of balance from a bipodalic position centered or alternated to a condition of monopodalism. The loads adapt to the variation of the supports reaching the peak when a single foot is in contact with the ground.

The curves show the effect during one step in the stance phase, considering the symmetry of the curves and the partial overlapping is simple to detect the total forces exerted during the full gait. For our scope, the principal component is the vertical one. During the use of the EXO, the system has to ensure the stability of the users and his stability. The current delivery needs to be high enough to sustain the total weight of users. The torque applied has to be equal and in much part higher than the moment due to the gravitation force to the body's centre of gravity.

1.3 Aim of this thesis

The thesis aims, consist to build, set up, and validate the prototype exoskeleton for lower limbs born from the collaboration between the IAS-Lab of the Department of Information Engineering and the Department of Industrial Engineering (DII), which was created to assist adults who suffer from severe motion conditions.

- During the lecture of this thesis, I will discuss about techniques and methods of assembly of the EXO's base frame. Introduction to the motorized joints with annexed harmonic reducer, and creation of the covers in plastics.
- I will explore the design and manufacture of design elements for user comfort and safety and to house essential components such as cables, drivers, power supply, supports, and sensors.
- I will deepen on the system control used to move and evaluate the response of the system. The choice of the sensing part with annexed code to be programmed.
- I will show the validation of the system by bench test on individual joints, and tests suspended on the single or entire limb.

Chapter 2

Design and construction

2.1 The wearable robots

2.1.1 Subdivision and classification

Inside the field of robotic systems, is difficult to cluster different devices into many groups. The number of research projects focused on constructing an integrated robotic exoskeleton system for assistance and therapeutic training has grown exponentially over the past decade. Many researchers worked on wearable robots and exoskeletons, introducing new concepts and innovations for supporting and training patients suffering from nerve impairments, loss of limb function, paralysis, muscle weakness, etc... that complicate the division.

The different types of wearable robots, such as the exoskeletons, can be divided by aim, actuation, design, and target of users. The most common strategy for classifying these systems is the current industrial scheme of project implementation, based on the material used, control method, signal sensing, user training, power transmission, mechanisms, the purpose of use, and actuated limb type[Meby Mathew, 2022]. This is not a standardization issue or some guidelines that the designer needs to follow but only a procedure to reduce the scattering of works already present and to focus on the actual challenge. The schematic result of the cluster is shown in Figure 2.1.

Purpose, a specific device can do excellent work when used in the right environment for the right issue. An assistant exoskeleton needs to help the users form the simplest movement to the most complex one during the normal routine life, ensuring security, and improving autonomy for the users. On the other side, the rehabilitation exoskeleton must re-teach the users how to move specific anatomic parts, encouraging the patient to complete the simplest and hardest movements. Assistant devices operate on users without the possibility of recovering the lost functionality of the body, while rehabilitation devices have to operate on users

with the possibility of recovering the damage.

Mechanisms, the disposition of the mechanical elements affecting the physiology of movements and the comfort of the users. The last prototype and available market devices present only a lateral mechanism. It's the most comfortable and easy maintenance choice, also that with lateral elements there are more configurations and design possibilities.

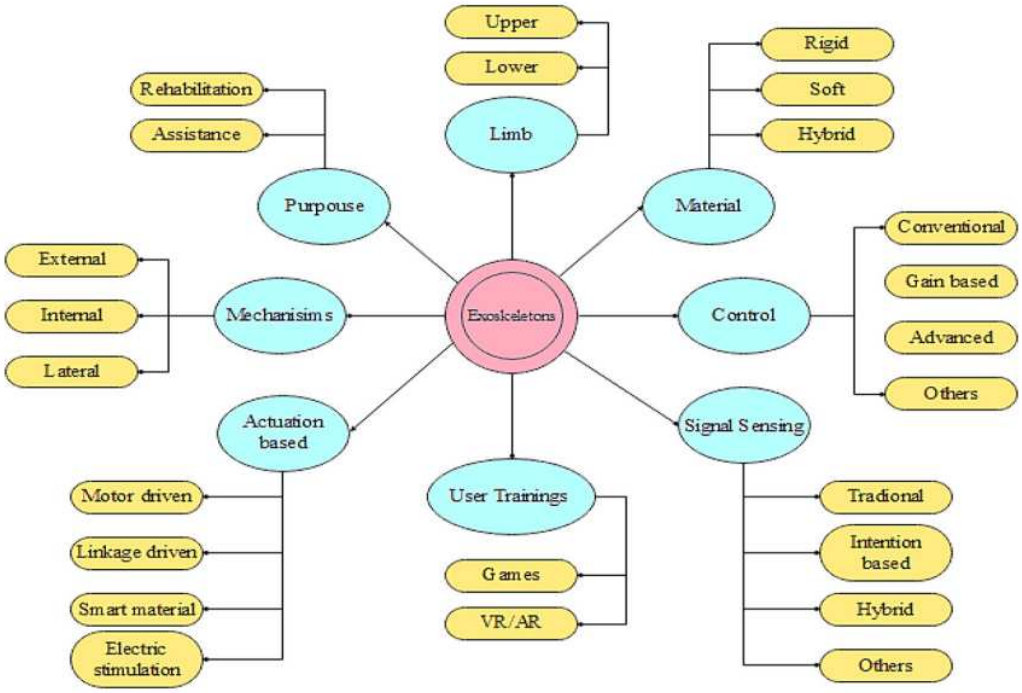


FIGURE 2.1: EXO wearable robot subdivision and classification [Meby Mathew, 2022].

Action and Actuation, proper actuation for the proper aim. There are active and passive exoskeletons. The active EXO works on people who need help in 75/85% of movement cases, the most common category is active exoskeletons, with 83% of the total market share. Actuators are present in active exoskeletons, which cause the user's body parts to move without the user exerting any energy. Passive systems are usually used for people who need light assistance, like during a long walk or to work without the possibility of injury. they don't have any type of actuators but are present strings and springs. The type of actuation changes by the type of assistance.

The main actuation modes that have been used recently in wearable devices are shown

as follows: hydraulic actuators, electrical motors, and pneumatic muscle actuators. Electric actuators are powered by any kind of electric motor, hydraulic and pneumatic actuators use pistons and soft actuators, and mechanical systems store or transmit mechanical energy through devices such as springs, dampers, pulleys, or gears.

Hydraulic actuators and pneumatic actuators were the first to be implemented. Unfortunately, this type of actuation is cumbersome, heavy, and inefficient. This led to the electric revolution, the implementation of a light electric motor with elevated torque to sustain the weight and to move the different parts. They can simulate the native articulated joint with an axial electric motor.

Material, the base of the device that characterizes strength and comfort. Among the various possibilities, there are rigid structures, usually made with heavy but very resistant materials; and flexible structures where you look for the meeting point between strength and the possibility of easing movement without impediment. The hybrid formulation is commonly taken into account to reinforce the structure where is needed and to reduce the weight in the last part.

Control, since exoskeletons are worn by humans, the interaction between the wearable robot and the human is a critical factor. There are different ways to implement the control for the EXO. The conventional closed-loop control/gain base compares the output feedback to the reference input signal and modifies the input accordingly. It's based on different combinations of proportional integral derivative (PID) gains. It's the most used for ensuring smooth and efficient control strategies. Of course, the actual challenges are to include artificial intelligence for shared control and high-level control using dynamic motion primitives together with LWR (EMG signal motion patterns are derived as dynamic motion primitives using locally weighted regression (LWR)) for learning the motion patterns of the subject.

Body part focused, exoskeletons can be designed for specific body parts, such as one hand, one leg, or the entire body. The most common body parts for exoskeletons are full-body, which assists all or most of the body; upper body, which involves the chest, head, back, and/or shoulders; and lower body, which involves the thighs, lower legs, and/or hips. There are also classes for specific limbs and joints, such as the knee, ankle, hand, arm, foot, etc. Additionally, there is a special class for any other exoskeleton that is not included in the previous classes. The most common body part-focused class is the lower body, which holds 56% of the total reviewed material in the cited article.

All the features discussed previously allow the designer to create a vast assortment of

devices with completely different scopes and actions. In the specific case of the exoskeleton for lower limbs, there is no doubt about the properties that it needs. A clear example of LLE can be seen in Figure 2.2.

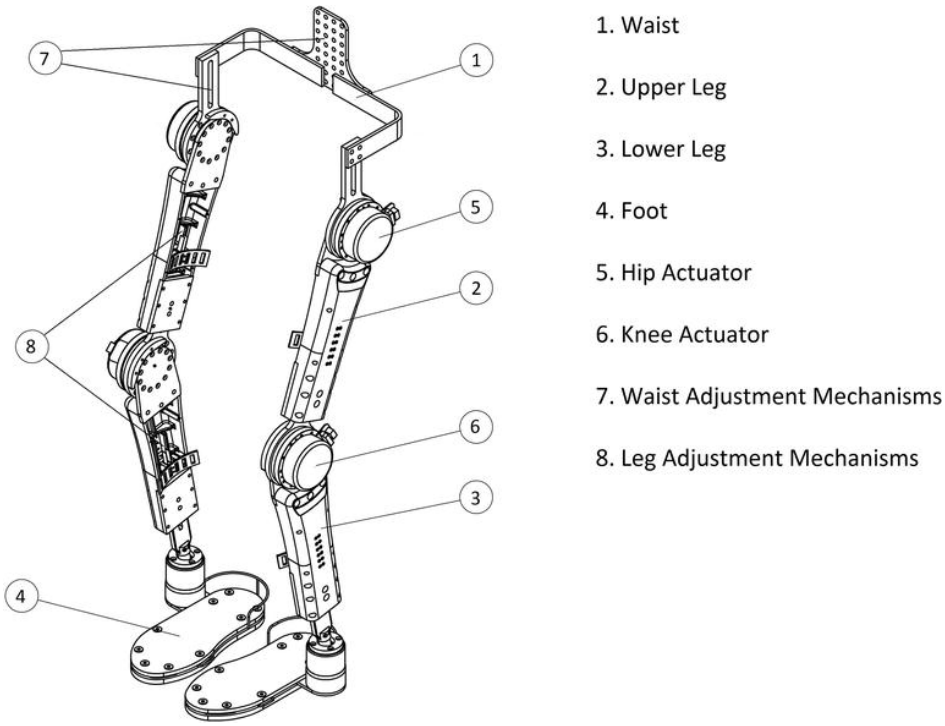


FIGURE 2.2: Representative draw scheme of general EXO for lower limbs [Önen, 2017].

It will be composed of a hybrid structure, rigid parts for the upper and lower leg, and a flexible waist part to improve movements during the walk and ensure the tilt of the native body.It will equipped with active and autonomous systems. The prototype that will be discussed soon has two actuators at the hip and the knee, they are electric axial motors that move respectively the upper and the lower leg. The power will be introduced by a portable battery on the back of the waist.

A device like this can be used for assistance purposes, for rehabilitation purposes, and as the main application targets of research or possibly clinical. This is a clear example and anticipation of what will be deepened in the following thesis

2.1.2 General architecture

The purpose of this chapter is to set out the main characteristics that the exoskeleton must have and then explain the various components in detail. The prototype of the lower limbs'

exoskeleton can be imagined as the result of three different layer overlaps, every single layer has a specific purpose, and was built taking into account the compatibility and functionality of the previous and next layers [Figure 2.3]. The three layers are:

- Base frame
- Motor and driver
- Cover and support

The point of departure was to design a structure to be compatible with men or women who suffer from severe motion conditions caused by stroke, spinal cord injuries, or trauma whose effect was the paralysis of lower limbs. For that reason, the prototype was thought to be adjustable for different types of anthropological measures and to be adaptable for many types of necessities.

Moreover, the structure was built to be used in outdoor environments, especially concerning agricultural works. That is an important indication of the choice of materials and the weight of the device. Overall, the structure was thought to be as simple and efficient as possible, reducing all the elements that are commonly considered an obstacle to the walk, and including all the features that allow a major movement comfort. That includes also the choice of degree of freedom of the robotic legs (DoF).

The number of DoF of the system is the sum of all independent movements (i.e., translations/linear displacements and rotations) that can be performed in all the joints of the device. An elevated number of DoF, like the number of the native legs, represents the condition of a utopian device, where all the motion conditions are available. Unfortunately, this is the most complex condition for the construction and control of the device. The design, the joint architecture, and the complexity of the actuators would be influenced by the DoF, and with that also the cost of the pieces.

As shown in the literature the robotics legs can have not less than 3 DoF and not more than 5 DoF, to ensure efficient control and comfort for users. In that specific case, the prototype has 3 degrees of freedom for the legs, one for each robotic joint. The movements that are allowed are essentially for walking along the sagittal plane, reproducing the flexion and extension of the articular joins and anatomical members [Monica Tiboni, 2022].

In this case, we distinguished into active or passive DoF. The term active degrees of freedom refers to movements that the user can actively control with the device, like walking

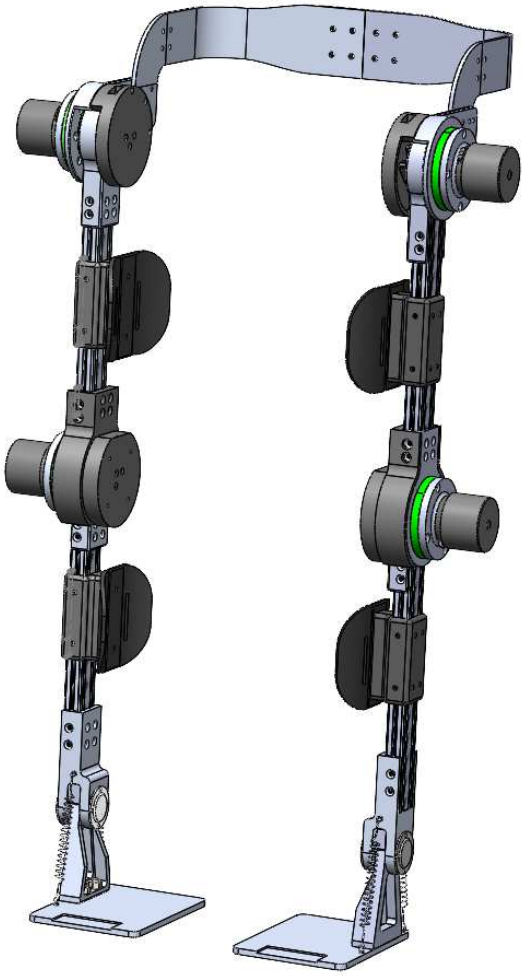


FIGURE 2.3: Lower limb EXO overview.

instead of sitting or standing up. The device's passive degree of freedom is the degree of freedom that can't be controlled by the user. The hip and knee joints are actuated by an electric motor, so they are active DoF, meanwhile, the ankle joint is a passive joint controlled by a couple of springs that can make plantar and dorsal flexion movements.

2.2 Elements material

This section will discuss the choice of materials for the different elements that make up the structure, showing the advantages and the disadvantages of the selected items; are also includes the cable management and the connection hardware.

Support frame

The base frame needs to be flexible and light, with a conservative cost, so the possibility is recluses to two applicants: steel and his alloys or aluminum alloys. Steel is the most important material in an industrial environment, the most used, and the most studied. It is strong and hard, easy to produce and manipulate for the specific dimension and application. But it's heavy and expensive. In the literature already exists EXOs with a main frame in steel alloy, but it was shown how the users suffered from the uncomfortable structure and difficulties of moving the members. That's because the steel structure doesn't permit the tilting movements of the pelvis or tiny flexion. Meanwhile, numerous pioneers in this field, using aluminum alloys with different concentrations of solutes, showed the effectiveness of a light structure that permits the tilting of the pelvis and reduces users's work during use. So, considering the main factors aluminium was chosen.

The choice of the specific aluminum alloy was very difficult to verify all the conditions of strength, size, weight, and flexibility, as well as cost, many were excluded. Falling back on the only possibility of guaranteeing a compromise between these characteristics. The aluminum in question is that of the 6082 series. The 6082 aluminum alloys are composed of aluminum-silicon and manganese, also called anti-chordal, and have excellent workability with machine tools and excellent weldability they are often used in shipbuilding naval, railways, and bicycles. Moving on to mechanical properties, 6082 T6 has a tensile strength of 310 MPa and a yield strength of 260 MPa. The density of this material is approximately 2.69 kg/dm3. All the components are made by CNC milling in aluminum alloy 6082, except the footplate and the lower part of the ankle joint which are made by Ergal 7075 aluminum alloy.

The main structure consists of extruded aluminum 40x20 mm profiles or finished elements at the lathe, with sliding inserts where an M5 screw can be screwed in, which can support the 80 kg weight of an average adult person. All of them were tested using the Finite Elements method. The axial forces and flexion forces values applied in the simulations were extracted from the walking phases of an able-bodied person.

Cover and support

All the feature elements present on the support frame are fixed by the covers or sliding guide. These elements are realized with 3D printing using PLA for industrial applications. The components were modeled to cover the joint, fix the legs of the users, and allocate the magnetic position sensor and motor driver necessary for motor control. The PLA was chosen for its tunability, durability, and easy manufacturing. It is known as common plastic, but it's

more strong and resilient. It's very light and cheap, and that guarantees the possibility to change some pieces when it occurs or to personalize the support made.

The exception concerns the leg supports, where the strips are located, that present carbon fiber to reinforce the PLA nucleus and prevent fatigue damage. These were produced with 3D printing techniques, with the rest of the covers, but covered with various layers of carbon fiber and glue.

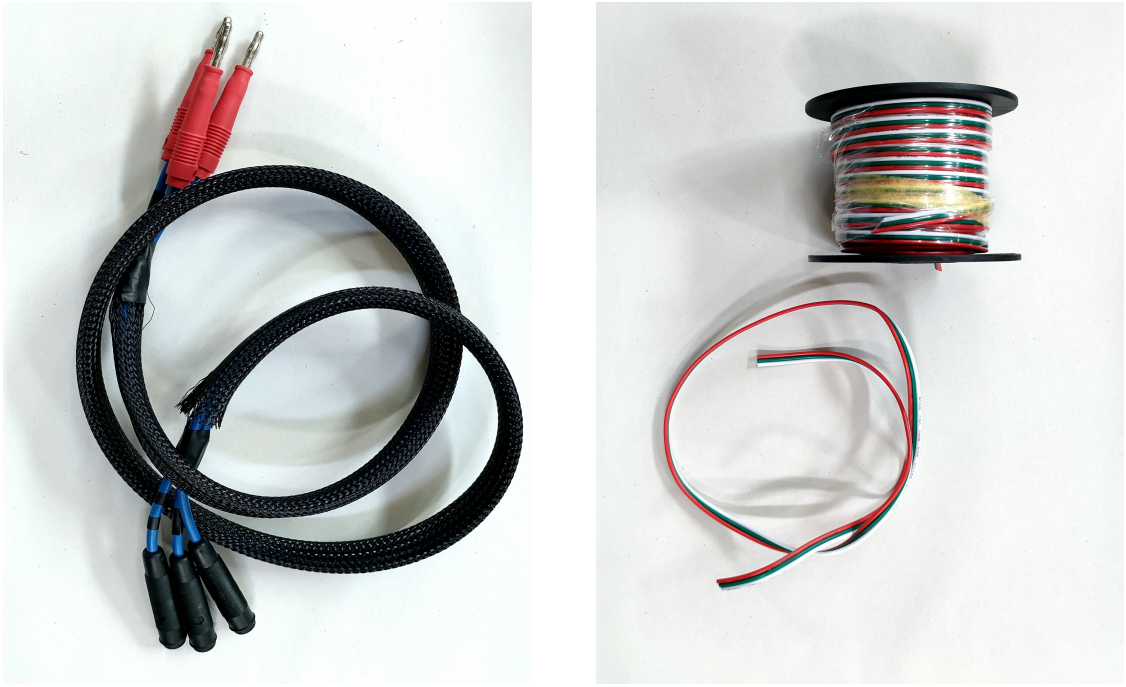
Cable and connection

For the wiring system we have purchased different types of cables divided by purpose:

1. Power cable: RS-PRO equipment connection cable, 4 mm², 12 AWG, 1 kV was used for the motor and driver power supply. It's a triple certification cable, also known as BS6231, panel wire, or cable for control groups. This cable is single conductor, flexible, class 5, compliant with BS EN 60228:2005. This means that the conductor is flexible with a core that is formed by several thin wires, which, in turn, are made of pure braided copper. The core is surrounded by HR PVC (heat-resistant PVC) insulation. This high-temperature, flame-retardant cable is commonly used in control applications, including instrument panels, engine control centers, and motor starters [Figure 2.4(A)].

2. Message cable: A 3-pin conductor Extension Cable was used for messaging and signal upload. It's a tinned copper wire with 3 conductors of 22 AWG (17 wires/wire) that has a rated voltage of 5 V and 200 V and an operating temperature of 80°C and 105°C, respectively. The connecting cables are designed to operate under strict conditions. Easy identification polarity for DC applications [Figure 2.4(B)].

All of these are complete with a plug connection. For the power cable, we used the same typologies of the motor module connection with three polarities. So we cut three pieces of RS PRO cable and adjusted the extremities with the connectors. In this case, we talk about the Multi-Contact Banana Connector. It's an uninsulated 4mm lantern-type connector and is used to join cables to electronic equipment. The lantern spring is made from a sheet of copper alloy to provide constant contact pressure and low resistance. It has a nominal current of 50 A. We used a male and female in the same wire for different extremities. The plug connectors for the message input-output were realized by Pitch Connector. There are a lot of types of connectors but we used the 3-pin connection for the magnetic sensor and the 6-pin (3+3) connection for the Hall sensors. They were made by hand using a crimping tool. We



(A) Power cable with connection. (B) Message cable for magnetic sensor and Hall probes.

FIGURE 2.4: Cable and connection hardware.

used the standard kit to make all.

To ensure the shielding of signal and the safety of the user all the cables present in the prototype were grouped inside expandable conduits for high-quality cables made of robust, environmentally friendly PET (polyester) monofilaments. Electrical wires and cables are simply passed through the sheath and the flexible structure of the sheath expands to accommodate a wide range of cable diameters.

2.3 Base frame

The base frame includes all the elements concerned with stability, support, and resistance. This is the nucleus of the device, where all the other elements will be equipped. From the upper to the lower part there are:

- Pelvis support
- Hip joint
- Knee joint

- Extruded profile
- Ankle joint
- Foot support

It is essential to balance the weight and flexibility of the structure, carefully choosing how to design the parts constituting the base of the device. Selecting the size and shape of the parts can increase the structure's stability during the walk phases, monopodalic or bipodalic support. These are the main conditions to verify the structure during use and ensure effectiveness and safety. Moreover, the flexibility of the structure ensures proper tilting of the limbs during the walking condition, reproducing the native movement of the legs and pelvis [Figure 2.5].

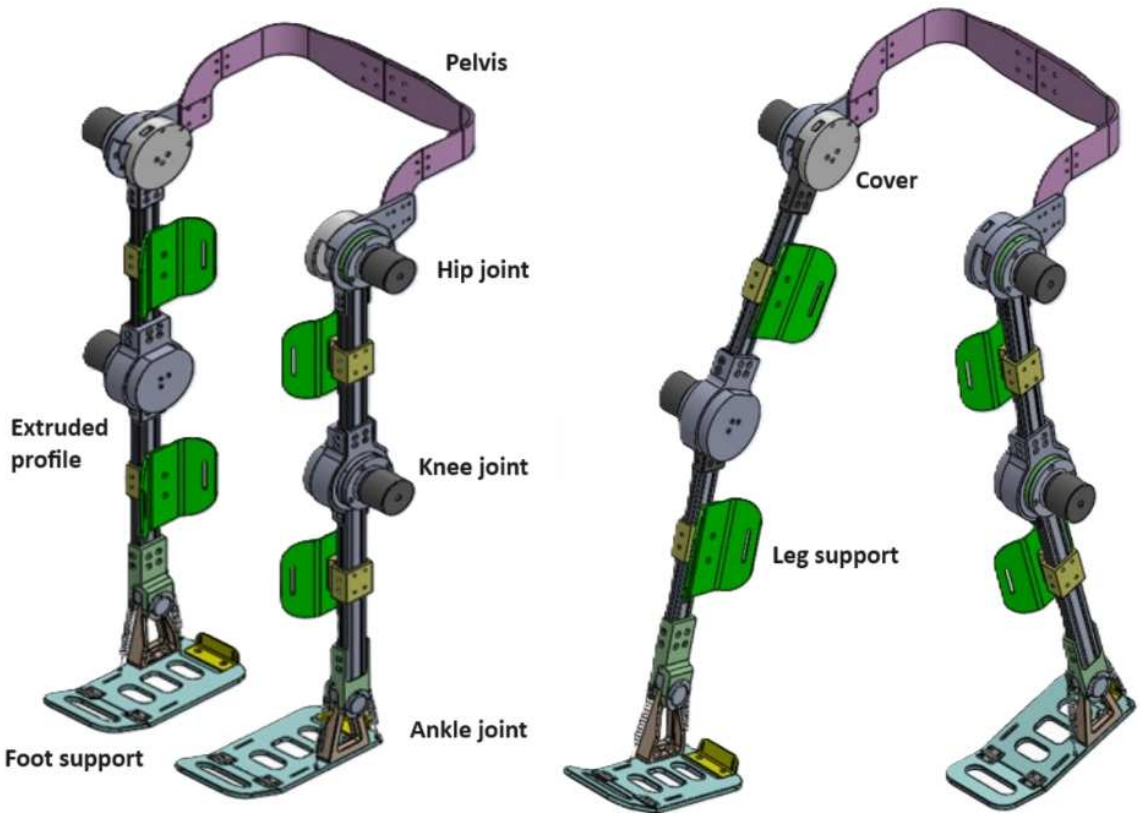


FIGURE 2.5: 3D CAD representation of all the elements equipped on the EXO.

The manufacturing is based on symmetric joins connected at the height of the hips by a curved structure that makes an angle of connection of 90°. All the elements were designed on

SolidWorks, tested in silico with a FEM simulator, and realized at the technical construction laboratory of DII.

2.3.1 Extruded profile

Extruded profiles are defined as interchangeable modules that ensure patient-specific comfort and use, are easy to produce, assemble, and define the connection between the joins. These profiles are adjustable structures, that ensure compatibility with different types of height. The dimensions of the different single tunable parts are related to the dimensions of the interesting anatomical part, which are: leg length, thigh length, foot size i.e., length between ankle and ground, and pelvis size [Figure 2.6].

FIGURE 2.6: Extruded profile on 3D CAD software.

The actual elements were chosen on average population data, there are many tables in the literature that report the dimensions of various bone segments, and we have chosen to use Dempster's tables which seem to be among the most reliable. So the images will always represent a structure tuned for users with a maximum height of 180 cm and a maximum weight of 85 cm.

2.3.2 Pelvis support

Pelvis support is an adjustable element, composed of 3 plates made of 6082 aluminum whose thickness of each of these plates is 5 mm. This system allows enlarging the back closure via an adjustable plate. The minimum opening size is 33 cm while the maximum is 45 cm. It is opened by moving the two sidebands and attaching them to the back plate by screws. The pitch of the holes is 12.5 mm. Also, on the side parts, holes are created to attach the camera in the future. The total weight of the pelvis plates is 3.03 kg [Figure 2.7].

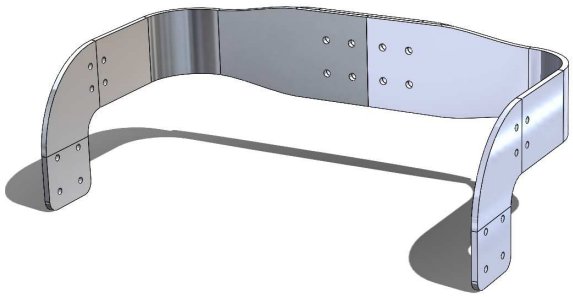


FIGURE 2.7: Pelvis support on 3D CAD software.

2.3.3 Joints

Knee and Hip joints

The connected parts are defined to create a cylindrical coupling with the axis of mutual movement orthogonal to the sagittal plane. The stator is the static piece connected with the previous member, the pelvis arm for the hip, and the upper leg for the knee, while the rotor is solid to the harmonic driver and drive shaft, and they are connected with the next member, the upper leg for the hip and the lower leg for the knee. The knee and hip joints are the only parts that are non-tunable. This is because the type of coupling between parts, which creates the mobile joint, is related to the size of the gearbox and the motor that will be housed. These elements are constrained in size and must ensure the correct transmission of torque to the next module [Figure 2.8][Figure 2.9].

Hip elements and weight		**Knee elements and weight**	
Motor:	0,56 kg	Motor:	0,56 kg
Junction flange:	0,132 kg	Junction flange:	0,132 kg
Gearbox:	0,98 kg	Gearbox:	0,98 kg
Pelvis connector:	0,277 kg	Thigh connector:	0,177 kg
Rotor coupling:	0,210 kg	Lower leg coupling:	0,164 kg

TABLE 2.1: Hip and knee elements and weight.

In the respective stators of the joins are present different types of dug finishes to ensure the movement of the rotor and to limit the excursion of the rotation. We get a maximum hip flexion of 95° and a maximum extension of 35°. While in the knee, the flexion is limited to

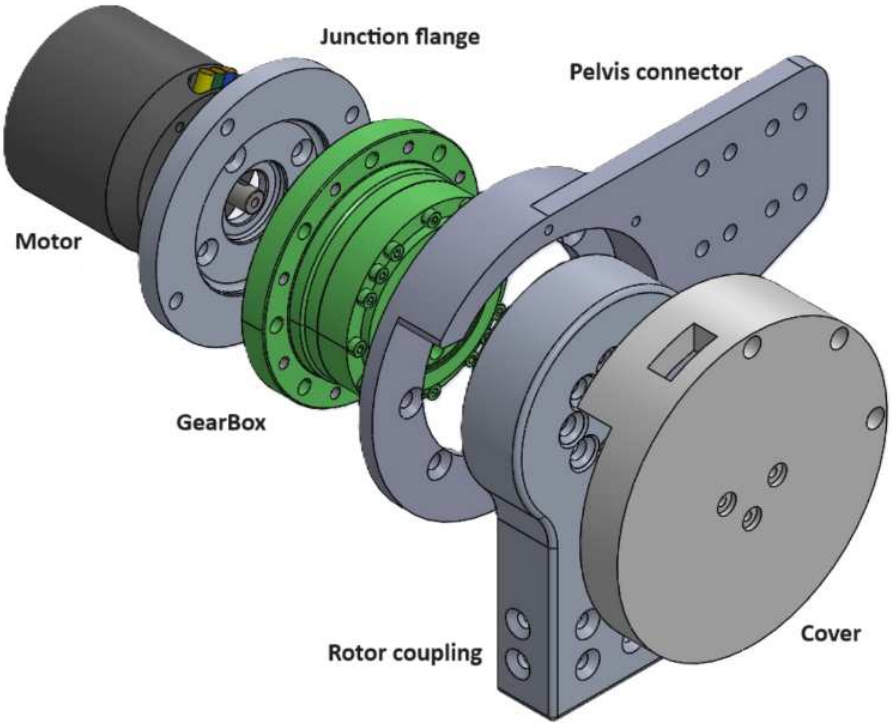


FIGURE 2.8: Hip joint on exploded view on 3D CAD software.

an angle of 90°, and the extension is until the leg reaches a vertical position, excursion of about 135°, this means that it will never have hyperextension. These angles are controlled by the programmed motor to ensure safety and stability during the walk but to be sure to avoid any kind of injuries to the users, the elements that compose the connection of joints are physically limited by rigid pieces.

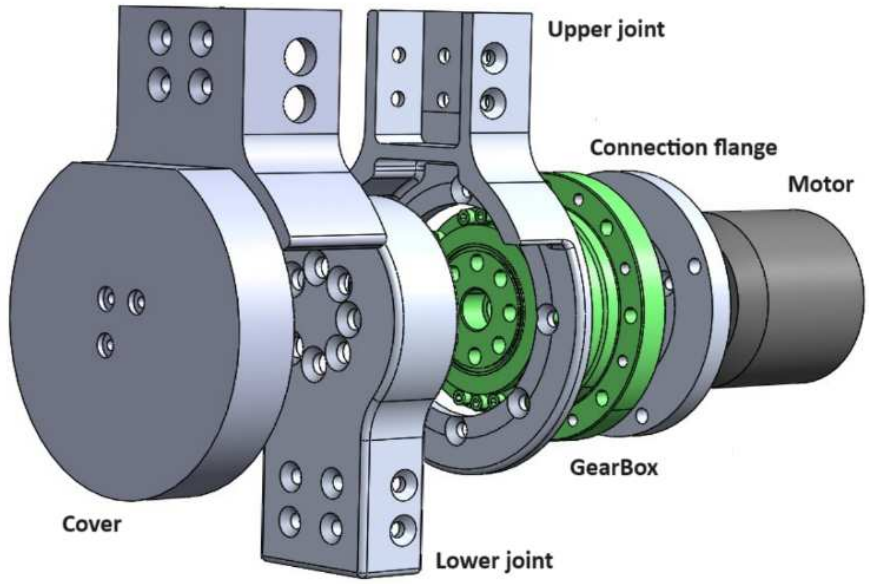


FIGURE 2.9: Knee joint on exploded view on 3D CAD software.

Ankle joint

The ankle joint follows the principle that remains the same as the cylindrical coupling. But in this case, there's a connection to ball bearings that ensures the movements of the parts along the sagittal plane. That's because there are no actuators in the ankle joints, only passive tools. The stator part is solid to the lower leg and is connected with the rotoric part with the bearing, meanwhile, it's connected to the footplate with springs. The closing system of the two connection modulus is interlocking. The foot consists of 8 basic elements [Figure 2.10].

Ankle elements and weight	
Upper part:	0,281 kg
Bearings:	2x 0,032 kg
Covers:	2x 0,011 kg
Shaft:	0,075 kg
Lower part:	0,206 kg
Foot plate:	0,615 kg
Mechanical limit switch:	0,031 kg
Spring:	0,05 kg

TABLE 2.2: Ankle elements and weight.

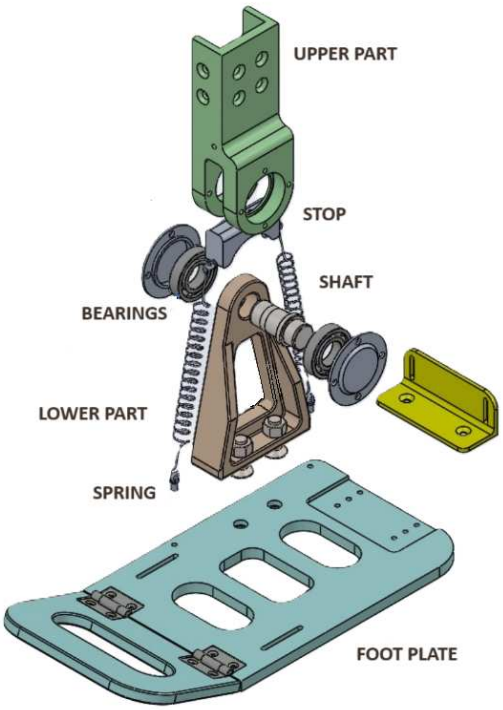


FIGURE 2.10: Ankle joint on exploded view on 3D CAD software.

The bearings are commercial elements selected from the SKF catalog. Specifically, SKF 16003-2Z. This type of single-row ball bearing with seals or dust shields, not to mention that this exoskeleton is designed for outdoor use, is particularly versatile, ensures low friction, and is optimized to reduce noise and vibration levels allowing high speeds.

As discussed previously, all the joints must include a limit of movement excursion, representing the physiological range of movement. In that case, the angle is the same in both plantarflexion and dorsiflexion, and it should be about 20° as suggested in the literature. Mechanically, the component that performs this function is an aluminum block, limit switch, that is placed in the lower part between the two ribs of the upper part and secured with a screw.

2.3.4 Foot support

Foot support is a fundamental part of the stability and effectiveness of the EXO during the walking phase. Many users can't control the foot during the walk, so it's necessary to block the foot and to ensure the discharge of the weight on the ground and not on the user's ankle. It was essential to improve a customized footplate with a passive return to zero position to

verify all of these conditions. This can be done via adjustable straps, adjustable supports, or other means of adaptation.

2.4 Motor and driver

2.4.1 FLIPSKY 6354 E-BOARD MOTOR

The Flipsky 6354 motor is a high-performance brushless direct current (BLDC) motor designed for applications such as electric skateboards, scooters, robots, and other light electric vehicles. This motor stands out for its efficiency, reliability, and versatility, making it a popular choice among DIY enthusiasts and electric vehicle manufacturers [Figure 2.11].

FIGURE 2.11: Flipsky 6354 motor frontal view form technical sheet.

The motor has a compact structure that allows easy mounting in tight spaces while maintaining a high power-to-weight ratio. The motor is designed to operate efficiently over a wide voltage range, offering flexibility for different battery configurations.

This type of motor has high performance and efficiency:

- No-Load Speed: Approximately 8600 RPM (revolutions per minute) at 10S (36V).
- Maximum Torque: 7Nm, ideal for rapid acceleration and hill climbing. The high speed and torque make the Flipsky 6354 motor suitable for applications that require dynamic and reliable performance.

Dimensions and Construction	
Outer Diameter:	63 mm
Body Length:	54 mm
Weight:	Approximately 800 grams
Motor Shaft:	8 mm with spline
Electrical Specifications	
Nominal Voltage:	10S-12S (36V-50.4V)
Maximum Continuous Current:	65A
Maximum Power:	Up to 2450W
Number of Poles:	14 poles (7 pairs of poles)

TABLE 2.3: Motor dimension and electrical specifications.

- Efficiency: Over 85%. High energy efficiency means that a greater percentage of electrical energy is converted into mechanical energy, reducing losses and improving the vehicle's range.

The Cooling System is integrated, and there is an internal ventilation system: the motor is designed with vents that promote cooling via forced ventilation, reducing the risk of overheating during prolonged use. This thermal design helps keep the motor operating at optimal temperatures, extending its useful life. The motor shaft is designed to fit a wide range of pulleys and wheels, making it easy to integrate into drive systems. Use of high-quality neodymium magnets and pure copper wire for the windings. The outer casing is made of anodized aluminum, resistant to corrosion and impacts. The quality of the materials used ensures high durability and consistent performance over time.

The Flipsky 6354 motor is ideal for a wide range of applications in light electric vehicles, including:

- Electric Skateboards: Its high power and torque allow for rapid acceleration and a good ability to overcome slopes.
- Electric Scooters: The motor offers a combination of speed and durability, essential for everyday use.
- Robotics: Thanks to its precision and reliability, it can be used in robotic applications that require precise and powerful movements.

The Flipsky 6354 motor is an excellent solution for those looking for a high-performance BLDC motor for light electric vehicle applications. Its advanced technical features, combined with robust construction and high efficiency, make it a prime choice for electric mobility and robotics projects.

2.4.2 LAIFUAL LSG-20 GEARBOX

Laifual LSG-20 harmonic reducer is a high-precision transmission device used in various industrial and robotic applications. This reducer uses harmonic technology to provide high reduction ratios, high torque capacity, and precision, as well as a compact and lightweight design. The harmonic reducer is especially suitable for applications that require precision and reliability, such as robotics, medical equipment, machine tools, and automation systems [Figure 2.12].

(A) Harmonic Gearbox 3/4 view, details on the screw holes.

(B) Harmonic Gearbox lateral view, details on the maximum dimension.

FIGURE 2.12: Laifual LSG-20 harmonic reducer 3/4 and lateral view from technical sheets.

Dimensions and Construction	
Outer Diameter:	60 mm
Body Length:	Approximately 37 mm
Weight:	Approximately 0.45 kg
Torque capability	
Rated Torque:	Up to 30 Nm
Maximum Instantaneous Torque:	Up to 60 Nm
Angular Backlash:	Less than 1 arcmin
Efficiency:	Greater than 80%

TABLE 2.4: Gearbox dimension and torque capability.

The LSG-20 reducer has a compact design that makes it easy to integrate into limited spaces, without sacrificing robustness and reliability. It's Available in various ratios, typically 50:1, 80:1, and 100:1. High reduction ratios allow for a wide range of applications

where significant speed reduction and torque increase are required. The ability to handle high torque makes the reducer suitable for applications with variable and intensive loads.

Extremely low angular accuracy ensures precise and repeatable movements, essential in robotics and advanced automation applications. The materials used are high-strength steel and special alloys for critical components, that ensure high resistance to wear and fatigue thanks to advanced heat treatments. The choice of high-quality materials and advanced manufacturing processes ensure a long operating life and stable performance over time.

The Laifual LSG-20 harmonic reducer is used in a variety of industries and technologies:

- Robotics: Used in robotic joints for precise motion control and accurate manipulation.
- Industrial Automation: Automated assembly systems, production lines, and manipulators.
- Medical Industry: Imaging equipment and surgical robots where precision and reliability are crucial.

Fundamental, for this type of technology, is to define the mounting interface. The harmonic reducer was built for standardized flanges, for easy mounting on various types of motors and mechanical systems. In this case, it was decided to build its interface. The design was thought on SolidWorks and the production was and the production was entrusted to the technical group. The versatile mounting interface facilitates the integration of the reducer into pre-existing systems or new designs. Fortunately, the output shaft was available in different configurations to adapt to various application needs, so there wasn't a need to do another adjustment.

Advantages of Harmonic Technology

- Compactness: Smaller and lighter design than traditional reducers with the same torque and reduction ratio.
- Reliability: Fewer moving components reduce the possibility of mechanical failure.
- No Backlash: Harmonic technology almost eliminates backlash, improving positioning accuracy.

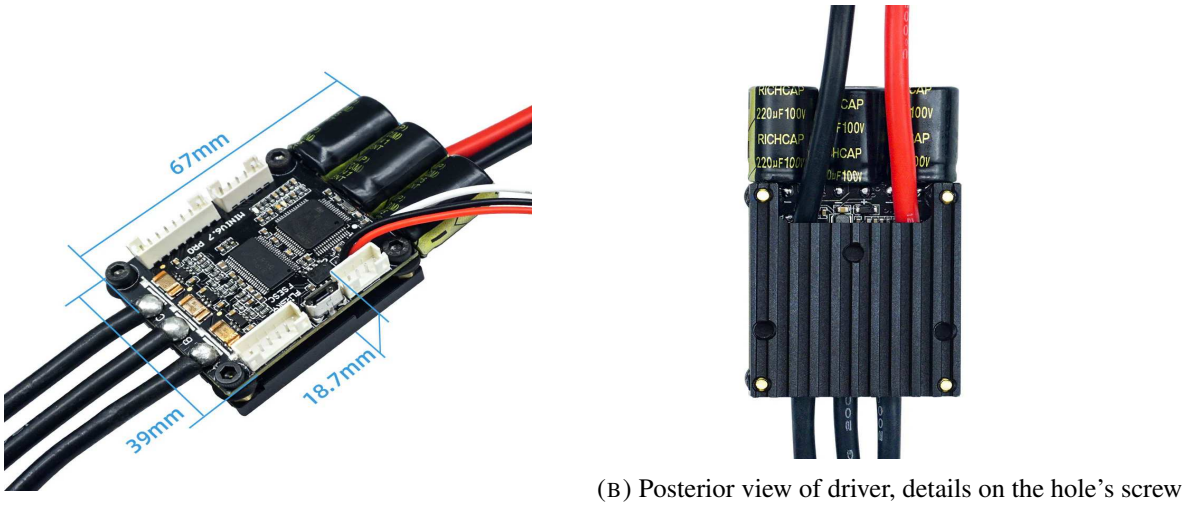
The Laifual LSG-20 harmonic reducer is a critical component for applications that require high precision, high torque capacity, and reliability. With its compact design, efficiency, and build quality, the LSG-20 is an excellent choice for improving the performance and precision of transmission systems in a wide range of industrial and robotic applications.

Dimensions and Construction	
Dimensions:	64 mm x 45 mm x 20 mm
Weight:	Approximately 110 grams
Electrical Specifications	
Supply Voltage:	8V - 60V (up to 14S LiPo)
Max Continuous Current:	70A
Peak Current:	200A

TABLE 2.5: Driver specification.

2.4.3 FSESC6.7 PRO DRIVER

The FLIPSKY Mini FSESC6.7 PRO, based on VESC6.6 technology, is an advanced electronic speed controller (ESC) designed for light electric vehicle applications such as electric skateboards, scooters, and robots. This ESC stands out for its high efficiency, high current handling capacity, and advanced control features, making it ideal for projects that require superior performance and reliability [Figure 2.13].



(A) Driver dimension form data-sheet.

(B) Posterior view of driver, details on the hole's screw to equip.

FIGURE 2.13: FLIPSKY Mini FSESC6.7 PRO from technical sheets.

The controller has a compact design that facilitates integration into tight spaces while maintaining robust thermal management thanks to the use of an anodized aluminum heat sink. The Mini FSESC6.7 PRO supports a wide range of voltages, offering flexibility for different battery configurations, and can handle high currents, making it suitable for applications with high power requirements.

The cooling system consists of a heat sink and, an anodized aluminum heatsink for better thermal management. The use of an advanced heatsink helps keep the ESC at safe operating temperatures, preventing overheating during intensive use. It has high-efficiency thanks to quality components and optimized design. The firmware is based on VESC Tool, it offers extensive configurability and customization. Although it has different control modes like sensored and sensorless, FOC (Field-Oriented Control), and BLDC (Brushless DC). VESC firmware allows you to customize the motor and system settings, improving performance and efficiency in different operating conditions.

Connectivity and Interfaces

- Communication Interfaces: CAN, UART, PPM, ADC, I2C.
- Telemetry: Supports real-time telemetry data collection and analysis.

Numerous communication options allow integration with various control and monitoring systems, facilitating diagnostics and performance optimization.

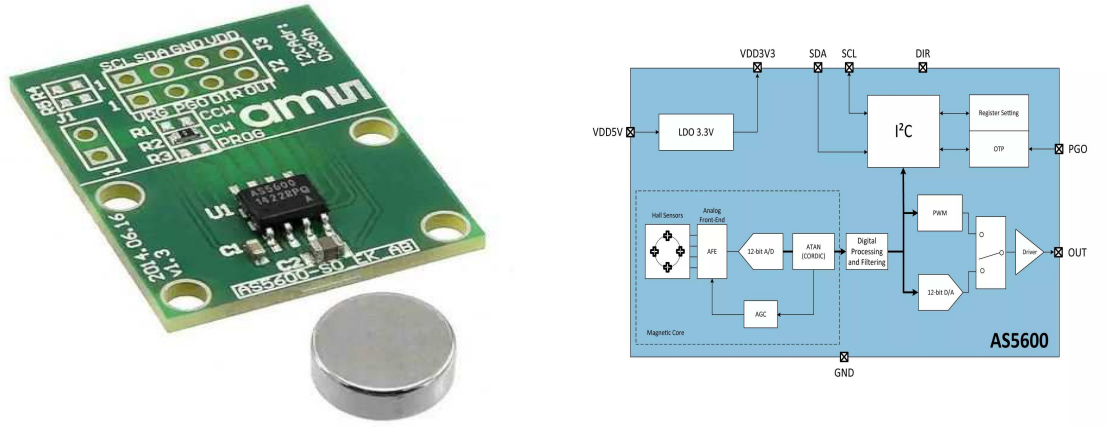
Compatible with a wide range of high-performance brushless motors and widely configurable via VESC Tool, suitable for various types of vehicles and custom applications. A wide range of motor compatibility and high configurability make this ESC extremely versatile for different applications.

The FLIPSKY Mini FSESC6.7 PRO is ideal for a wide range of light electric vehicle applications. It is a high-performance ESC controller that combines flexibility, reliability, and advanced features. With its high current handling capacity, compatibility with a wide range of voltages and motors, and numerous configuration and protection options, this ESC is an excellent choice for electric mobility projects and industrial applications. Its build quality and advanced technical features make it an indispensable tool for those looking for performance and reliability in their electric propulsion systems.

2.4.4 MAGNETIC ENCODER

AS5600 sensor is a high-precision magnetic encoder, designed for non-contact angle sensing. This sensor is widely used in applications that require precise rotation angle measurement, such as robotics, motor control, actuators, and automotive applications. AS5600 sensor is appreciated for its ease of integration, high accuracy, and robustness. Based on a Hall sensor to detect the angular position of the magnetic field. Non-contact operation reduces mechanical wear and increases the life of the device [Figure 2.14].

Low operating voltage and low power consumption make it suitable for low-power applications. The wide measurement range and extreme temperature capabilities make it versatile



(A) Sensor chip and magnet (B) Eletrical data sheet scheme

FIGURE 2.14: Magnetic encoder AS5600 images form data sheet

Electrical Specifications	
Supply Voltage:	3.0V - 3.6V / 5.0 - 5.5V
Current Consumption:	Approximately 6.3 mA in operating mode
Output Interface:	PWM (Pulse Width Modulation), I²C
Operating Range and Limits	
Measurement Range:	0° to 360°
Operating Temperature:	-40°C to +125°C

TABLE 2.6: Magnetic Encoder specification.

for various industrial and environmental applications. the resolution is up to 12 bits, corresponding to 4096 discrete positions per complete revolution, and the accuracy is about ±0.5 degrees, providing highly accurate angular measurement. High resolution enables precise and detailed measurement of angular movement.

This encoder is extremely rugged and reliable, it has noise immunity and high immunity to external magnetic disturbances, thanks to its compact structure and, moreover a long operating life due to the absence of mechanical contact. The rugged design and immunity to external noise ensure reliable performance even in harsh environments.

It supports automatic calibration to accommodate various magnetic configurations. Self-calibration capabilities simplify integration and improve accuracy without the need for complicated manual configurations. It's compatible with standard I²C interfaces for easy integration with microcontrollers and other devices and works with standard diameter magnets. Compatibility with standard interfaces and magnets facilitates implementation in various existing systems.

The AS5600 sensor is used in a variety of applications that require precise angle measurement:

- Robotics: Used for position feedback in robotic joints and actuators.
- Motor Control: Used for switching and precise position control of brushless motors.
- Actuators and Valves: Used for position monitoring in linear and rotary actuators.

The AS5600 sensor is a versatile and highly accurate magnetic encoder, ideal for a wide range of industrial and commercial applications. With its high resolution, accuracy, robustness, and ease of integration, the AS5600 is an excellent solution for angle sensing in advanced motion control systems. The non-contact technology improves the sensor's life and reliability and provides consistent performance in harsh environments, making it a critical component for applications that require precision and reliability.

2.4.5 ARDUINO MEGA AND SHELL

The Arduino Mega 2560 is a development board based on the Atmega2560 microcontroller from Microchip Technology. It is particularly appreciated for projects that require a large number of digital and analog inputs/outputs and for its ability to handle complex operations thanks to greater memory than other Arduino models.

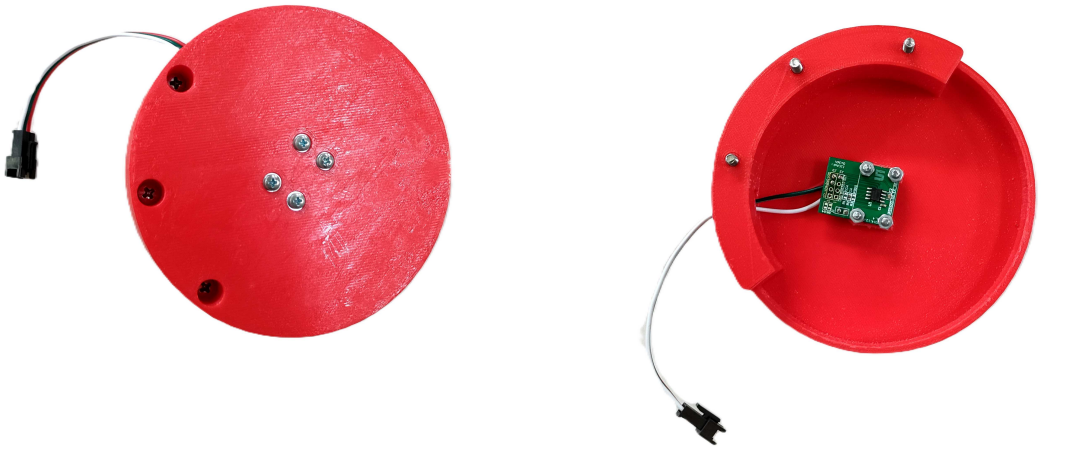
Electrical Specifications	
Operating voltage:	5V
Input voltage (recommended):	7-12V
Input voltage (limits):	6-20V
I/O Digital Pins:	54
Analog Input Pins:	16
DCt for Pin I/O:	20 mA
DCt for Pin 3.3V:	50 mA
Flash Memory:	256 KB

TABLE 2.7: Arduino specification.

The Arduino is fundamental to interconnect all the sensors with the microcontroller. To ensure the application of four sensors the base platform was upgraded with the Arduino Shield, a plate-like electrical plate thousand holes for the cable bridge.

2.5 Cover and support

All the elements added to the base frame are defined as "covers" to guarantee the user's safety, block the user's extremities, and introduce the magnetic encoders. These are hip joint covers, knee joint covers, thigh support, lower leg support, and driver support. As said before they are all made by PLA.



(A) Front view (B) Posterior view

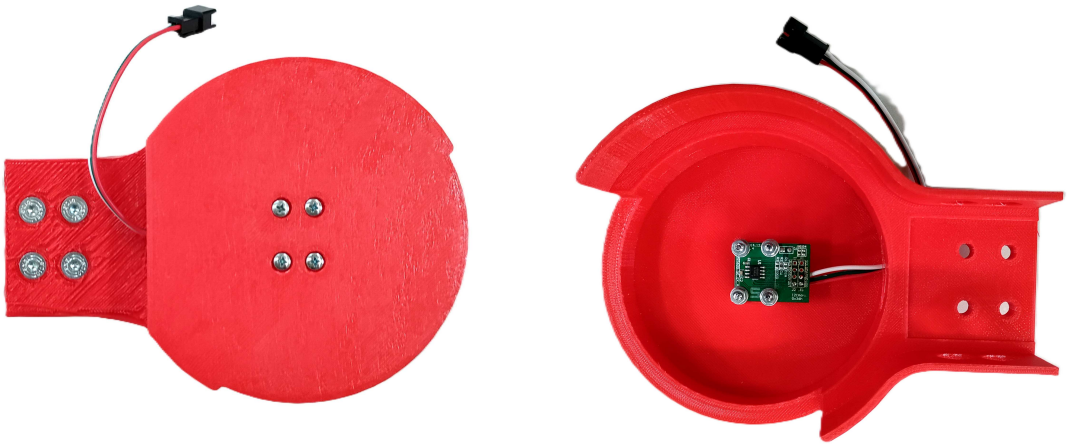
FIGURE 2.15: Hip covers with magnetic sensor and connection cables.

2.5.1 Joint covers

Starting with the maximum dimension and the geometry of the elements to cover, we designed a 3D model to define a safe and roughness-less surface to ensure comfort and prevent any injuries from the rotary parts. It was made particular attention to the magnetic insert. The joint covers have the specific task of supporting the magnetic sensor during the movement. The prototype of the design was done on SolidWorks, tested with FEM, and then the product was made through rapid prototyping systems, such as 3D printers at AIS-lab. The insert for the magnetic sensor consists of two-width columns with holes for the four screws [Figure 2.15][Figure 2.16].

2.5.2 Leg supports

The leg's supports are made of three parts: a sliding guide, a parabolic panel, and a connection plate. The leg's supports are divided into two categories, the upper leg support, for the

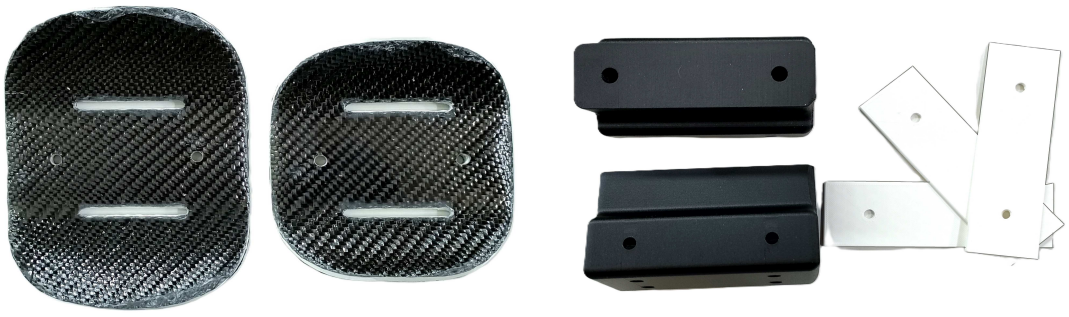


(A) Front view (B) Posterior view

FIGURE 2.16: Knee covers with magnetic sensor and connection cables.

thigh, and the lower leg support, for the calf. The difference between the types consists of the parabolic panel, for the thigh we chose a large panel, meanwhile for the calf, the panel is less big [Figure 2.17].

The three parts are connected with two screws, the screws are screwed into two tiles. The tiles in question are self-expanding threads during screwing, creating a lasting block between the screw and the base. To introduce these tiles we used the hot forcing technique, thanks to an industrial dryer.



(A) Carbon fiber reinforced plate. (B) Items essential to equipped the strings.

FIGURE 2.17: Printed supports for fixing the legs.

2.5.3 Driver supports

Four actuated joints deserve four different motors with their drivers. The four drivers will be connected to the motors by three power cables, a 6-pin connector for Hall sensors, and two power cables to the battery power supply [Figure 2.18].

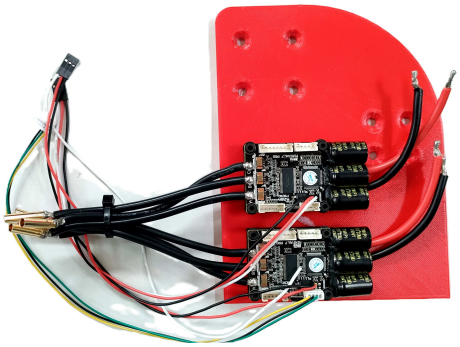


FIGURE 2.18: Drivers equipped on a printed plate for support.

The positioning of these elements cannot be random. We had to ensure to not disturb the movement of the members during the walk, so the cables, to get behind the hips without an obstacle, must pass through the sides of the members. The best strategy position was to equip the driver to the side of the pelvis support with two shaped plates. The plates were designed on Solidworks, and the shape was designed to fully adhere to the pelvis support profile. Also, the holes were defined as to consent the use of countersunk screws. The driver was blocked by three blind screws.

2.6 Assembly

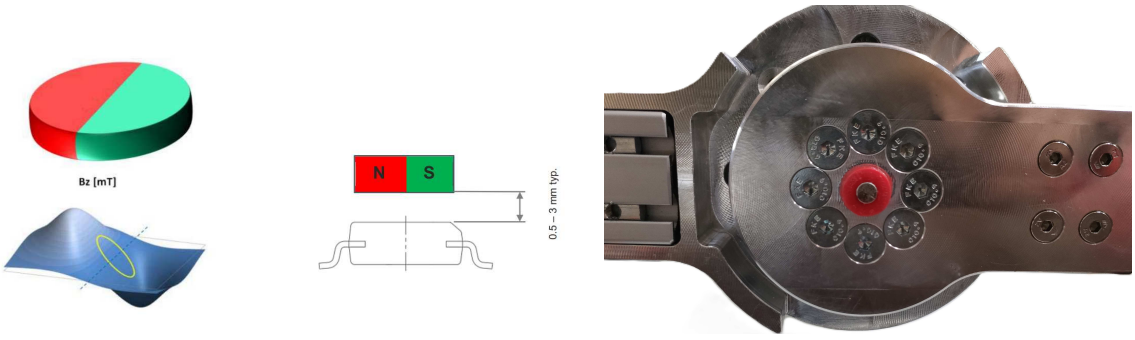
This section is dedicated to the methodology of assembly of components, highlighting specific procedures or paying attention to some choices taken into account.

- All the extruded profiles are linked with the three joints with dowels and countersunk head screws M5. The tiles used are self-locking, and the materials are industrial steel, such as the screws.
- The joint is complete when the motor, the gearbox, the interface plate, the two cylindrical parts, and the self-expanding bearing of connection are all linked together at the same time. The assembly scheme is relatively simple. The motor and the reducer are connected via the junction flange, through a threaded connection, the flange will be

the same size as cylindrical parts. The gearbox has the function of reducing the revolutions of the shaft coming out of the motor. The rotation of the lower joint is limited by a mechanical limit switch located on the upper joint. Inside the gearbox will be the deposition of a layer of industrial application fat [Figure 2.19].

FIGURE 2.19: Gearbox and connection plate with industrial fat ready to assemble.

The ankle joints were assembled by a pressure piston, that ensures the shaft with the two bearings and the upper part, the clamping is impervious to reverse and cannot be undone.



(A) Specific disposition for the correct location of the sensor and magnet.

(B) Focus view on magnet and support for the knee joint.

FIGURE 2.20: Magnet and printed support technical information and location.

The magnetic sensors are lodged into the internal part of the covers, and the magnets are glued to the rotary parts of the joins with a disc support. We build a specific element for the tightening of the magnet [Figure 2.20 (B)]. The AS5600 requires the magnetic field

component Bz perpendicular to the sensitive area of the chip. Along the circumference of the circle of the Hall element the magnetic field Bz should be sinusoidal. The gradient of the magnetic field of Bz along the radius of the circle should be within the linear range of the magnet to eliminate the displacement error using the differential measurement principle. The typical air gap is between 0.5 mm and 3 mm and depends on the magnet selected. A larger and more powerful magnet allows a wider air gap. Using the AGC value as a guide, you can find the optimal air gap by adjusting the distance between the magnet and the AS5600 so that the AGC value is at the center of its range [Figure 2.20 (A)].

Chapter 3

Control architecture

3.1 General known of control

The fundamental part of a robotic device is the control system. The joints can be seen to be manipulated by an automatic device (control unit) that uses a processor. The command is typically information detected by the control unit from the user's intention, that needs to be converted to the most suitable physical size to report the imposed commands to the drive drivers. The driver is the information transducers that convert digital electric impulses into adjustable electric impulses at a given voltage for actuators. The power source provides the energy. Based on the indications from the human-machine interface (HMI) and the sensors' data, the control unit decides how the actuators must be powered. This process is fundamental and implies the creation of some constraints such as the maximum voltage of the information sent that will be, for common applications with electric motors, 10 volts in positive or negative [Monica Tiboni, 2022].

The electric actuators come in different types and sizes and are controlled by the control unit. Brushed DC electric motors are the most widely used among the different types, followed by brushless BLDC motors. The brushed DC motor is used, in many cases, because of the simple structure of the motor itself and its electronic drive; however, the brushes and commutator system require regular maintenance. Therefore, the main advantages of brushless vs. brushed motors are greater reliability, due to the lack of brushes and better dynamic performance, allowed by a lower rotor inertia and a higher power-to-weight ratio. In some cases, direct drive torque motors, which are placed at the joints, are used [Figure 3.1].

In cascade to the process, there will always be sensors that detect the position of the motors corresponding to the joints and verify the error between the given information and the action performed. This is done for each defined time of sampling and continuously to monitor and if appropriate correct an action in progress or yet to be performed. Sensors are an indispensable element in an exoskeleton, as they allow the detection of information from

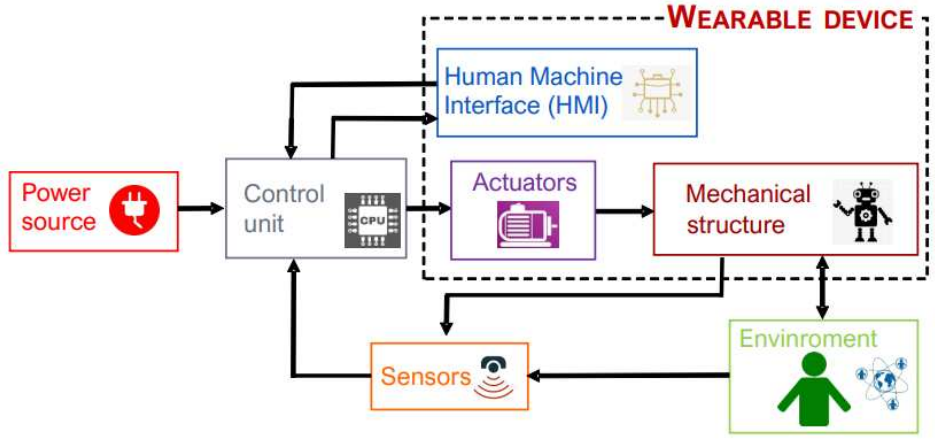


FIGURE 3.1: Block scheme of generic robotic control system [Monica Tiboni, 2022].

the environment (that includes the user) with which the robot interacts. Inside this general section are included also the encoders. The angular position or motion of a shaft or axle is converted to analog or digital output signals by a rotary encoder, also known as a shaft encoder. Rotation angle signals are generated by a shaft encoder in the form of pulses, which are then processed by a signal conditioning system to produce a more usable signal.

Most control systems are automatic, requiring few operator input parameters essential for a specific task, in some cases like intelligent systems, it may not be necessary. That's the case of the control then acting regardless of the operator when it has received a task and monitors the whole process using a sampling rate usually equal to 1000/1500 Hz. Usually, the control system can be defined as a complex algorithm that elaborates the input data from the user's command or the environment and applies specific laws and filters to convert them into electrical information compliant for the motor driver. The algorithm law is based on what we want to obtain, for example, if is our desire to move the right leg forward, the control unit has to coordinate two joints to reproduce the step. To do this the control unit needs to send to the specific joint a function that represents the position profile of a native joint like hip or knee. Given that, the actuator needs electrical input (a voltage), the correct move must be explained precisely and quickly by the control unit after converting an understandable position profile into an electrical pulse. Corrective or control actions must be swift and safe, and to ensure safety the movement needs to be constantly under control.

Generally speaking, before arriving at an efficient control described above, we find more or less simple solutions according to the cases. The types of control are traceable to the

following list where each step corresponds to an upgrade of the previous step:

1. Open loop: A control system where the output is not fed back to the input.
2. Closed loop (feedback): A control system where the output is fed back to the input for continuous correction.
3. Closed loop (feedback) and direct action chain (feedforward): present, in addition to feedback, a direct forward action; are used in cases where the specifications are very pushed.

3.1.1 Open Loop Control

In control theory, an open loop controller, also called a feedback-free controller, is a control loop part of a control system in which the control action is independent of the "output process", which is the process variable being controlled.[Figure 3.2] The static torque-speed characteristics of the converter+motor system are compatible with those of the load and intrinsically stable. The fundament components are:

1. Input: Command or desired state.
2. Controller: Processes the input and generates an output.
3. Actuator: Executes the command from the controller.
4. Output: Result of the actuator's action.

Is a no-feedback mechanism. Simpler and less expensive. Generally less accurate due to no error correction. Suitable for systems where the relationship between input and output is well understood and stable.

A classic example is the perpetual escalators that adapt only the torque to the load to maintain only the constant speed or washing machines (older models), simple lighting systems, and toasters.

Advantages	**Disadvantages**
Simplicity and ease of implementation	Lack of error correction
Lower cost and complexity	Poor adaptability to changes or disturbances
	Limited accuracy and reliability

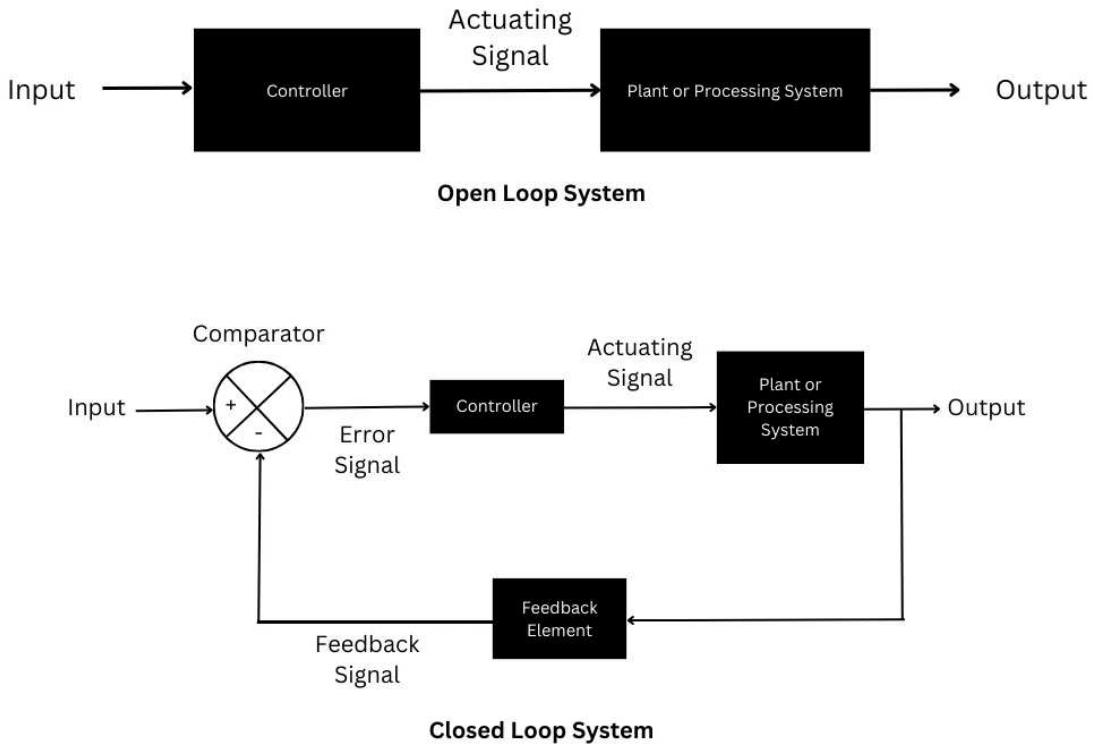


FIGURE 3.2: Open Loop and Closed Loop [1].

3.1.2 Closed Loop Control

The closed-loop control (or retracted or backward or feedback), more complex but much more flexible than the first, can make a system stable that in itself is not at all. In this case, the control ring returns, at the input of the process that you want to control or make stable, a function of the output that must be algebraically added to the signal already present at the input.[Figure 3.2] The fundament components are:

1. Input: Command or desired state.
2. Controller: Processes the input and generates an output.
3. Actuator: Executes the command from the controller.
4. Sensor: Measures the actual output.
5. Feedback Path: The output is fed back to the controller to compare with the input.
6. Output: Result of the actuator's action.

Advantages	Disadvantages
Improved accuracy and reliability	Higher cost and complexity
Continuous error correction	More components and potential points of failure
Better adaptability to changes or disturbances	

Incorporates a feedback mechanism. More complex and potentially more expensive. Higher accuracy due to continuous error correction. Suitable for systems requiring high precision and adaptability to changes. Much more flexible and robust than changes in process parameters and torque disturbances;

3.1.3 Overview scheme of EXO control

The control type used in the following thesis is the closed-loop control. Our research aims to build a total functional EXO, in this first phase, the EXO will be controlled by an external operator. To achieve this objective we started with a micro-controller connection and laboratory power supply to study and set up all the essential features for the goal.

The previous graft, Figure 3.3, shows the actual scheme of connection and node of control for the EXO of lower limbs. As it's possible to see, all the joins have a power supply connection, encoders of position, Hall sensors, one driver, and a connection with the microprocessor, or control unit, thanks to a dongle.

The control unit acts as the master node, and we use a portable microcomputer to program and control the motor drivers. Below the master node, there is the node to the serial communication with sensors and the node to the communication with the drivers. A node is a fundamental unit that represents a process that performs a specific task. Each node can communicate with other nodes to exchange information and coordinate the actions of the overall robotic system. The environment of implementation used is the Robot Operating System (ROS) and the operated program implemented to control the joins is the SimpleFOC vector controller.

The power supply, the feedback of Hall sensors, and the control input are all mediated by the drivers. A single driver can act for a specific motor, while for the magnetic encoders, the choice was to use an Arduino to group the output of the magnetic sensor. That simplifies the communication with the master node, which needs only a singular serial connection for all the encoders.

The current control process consists of three operational sequences, consequential and interconnect, included within the SimpleFOC control software. They are:

- The low-level control

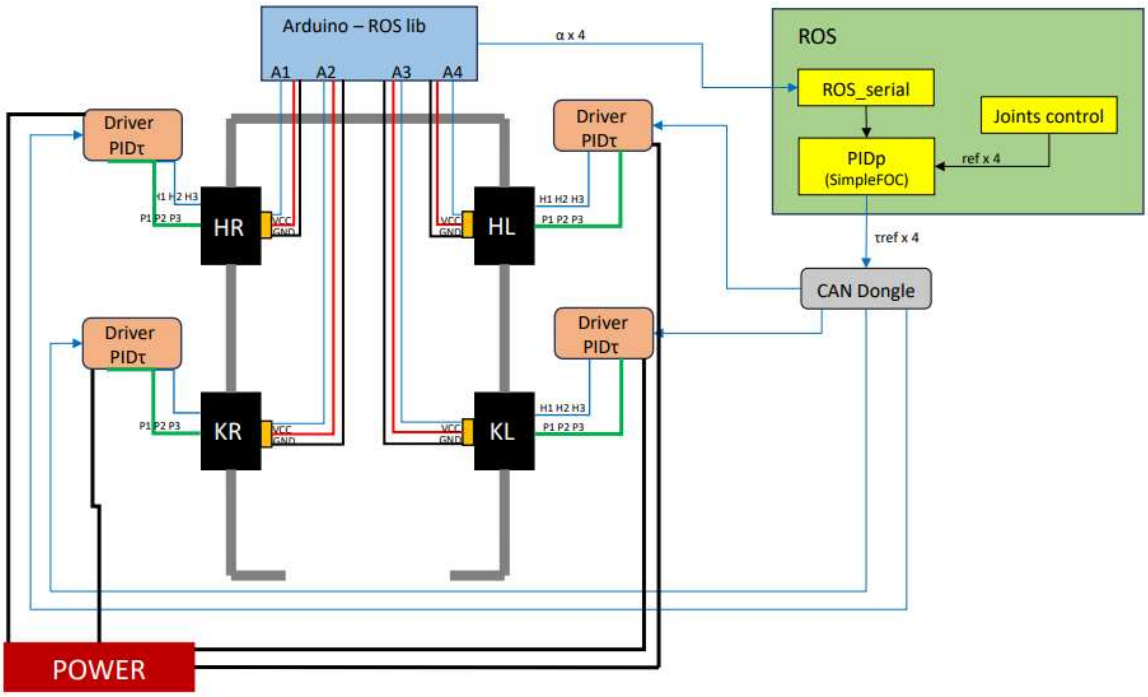


FIGURE 3.3: Overview scheme of EXO prototype control. Black line: Voltage connection between drivers and power or encoders' ground. Blu arrows: Serial cable to control communication. Blue line: Bridge cable for the connection of Hall sensor and drives. Green line: Bridge cable for power supply. Red line: Voltage connection between the encoders and Arduino. Light blue line: Analogic connection output for the encoders.

- The mid-level control
- The high-level control

Low-level control takes care of the direct implementation of motors and sensors (such as position, speed, and torque control. The method of control is summarized into two control ring actions. The internal ring is concerned with the torque/current control using the Field Oriented Control (FOC) [Antun Skuric, 2022]. While the external control ring is concerned with the position and velocity motion control. Inside the two rings, there are respectively two PID-implemented control methods.

The second step consists of establishing the target current to introduce as the first input of low-level control, based on the profile of position and velocity control. This takes part in what is known as mid-level control. At last, a high level of control. In robotic systems, such

as exoskeletons, it concerns the planning, coordination, and supervision of system movements and actions. High-level control focuses on more abstract and complex aspects, such as motion planning, path generation, stability control, and user interaction.

3.1.4 SimpleFOC library

Field-Oriented Control (FOC), also known as vector control, is a technique used in variable frequency drives to control AC motors, particularly induction motors and permanent magnet synchronous motors (PMSMs). SimpleFOC [2] implements the FOC algorithm routines, motion control strategies, generic hardware interfaces, and various configuration parameters encapsulated in an object-oriented C++ library. It provides the users with an intuitive way to develop their motion control application and the possibility to change all the hardware components (motor, sensors, drivers, microcontrollers) with relatively minor code modifications. FOC provides precise control of motor torque and speed, akin to that of a DC motor, by controlling the stator currents in such a way that the magnetic field generated by the stator is always perpendicular to the rotor field.
FOC is a well-known example of one of the most efficient techniques developed over the years. However, the FOC approach has a complex control architecture that requires significant computational performance.

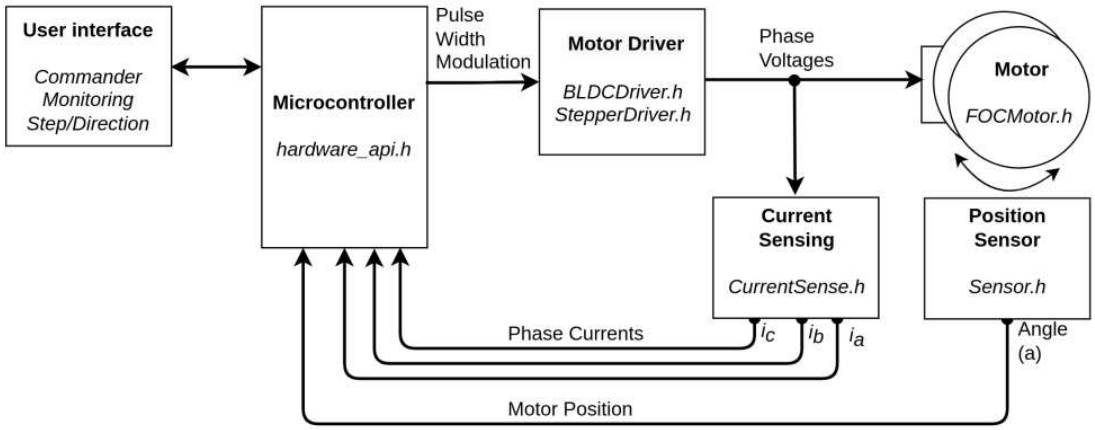


FIGURE 3.4: SimpleFOC modular architecture overview [Antun Skuric, 2022].

Key Concepts Motor:

- Induction Motors, the stator creates a rotating magnetic field that induces a current in the rotor, causing it to rotate.

[2] https://simplefoc.com/

- Permanent Magnet Synchronous Motors, the rotor has permanent magnets, and the stator windings create a rotating magnetic field that interacts with the rotor field to produce torque.
- Field-Oriented Control, FOC decouples the control of the motor's magnetic field (flux) and torque, enabling independent control similar to that of DC motors.
- Reference Frame Transformation (Park and Clarke Transformations) These mathematical transformations convert the three-phase stator currents (a, b, c) into two orthogonal components in a rotating reference frame (d, q).
- Control Loops:
  - Current Control Loop: Regulates the d-axis (direct axis) and q-axis (quadrature axis) currents.
  - Speed/Torque Control Loop: Adjusts the reference for the q-axis current to control motor torque and speed

SimpleFOC provides a modular implementation of the FOC control architecture by intuitively dividing the code into generic blocks representing different hardware and software components, such as motors, drivers, position sensors, current sensing, user interfaces, and finally the microcontroller-specific code.

Each block represents the operational necessities of the FOC and motion control, Figure 3.4, as well as all the initialization, calibration, and communication functionalities specific to the hardware. For each of the generic blocks, SimpleFOC provides multiple cross-platform implementations based on different hardware specifications and in this way supports many motor, sensor, driver, and microcontroller combinations. The goal of this modular architecture is not only to facilitate the prototyping and design process for users but also to allow for easier integration and support of new hardware components.

3.1.5 PID

The torque is the principle parameter to tune to realize a complete low-level control and complement the current. To establish the torque necessary to improve the right movement we have to understand the force acting.

The product between the moment of inertia and the derivative of the angular displacement will give as a solution the total torque developed or necessary.[Figure 3.5] This can be studied as the contribution of three elements:

- Torque needed to start and maintain the bike;

$$I_m\ddot{\vartheta}(t) = C(t) = C_m(t) - F_m\dot{\vartheta}(t) - C_r(t)$$

$$C_m(t) = I_m\ddot{\vartheta}(t) + F_m\dot{\theta}(t) + C_r(t)$$

FIGURE 3.5: Torque equilibrium equation for motorized joint.

- The contribution of viscous friction, dependent on speed;
- Resistant torque, linked to all resistive phenomena that occur with motion or even only to gravitational forces.

All rates are dependent on $\theta(t)$, and are directly calculable, while the resistant torque is the only one not known a priori. In some cases, this can directly coincide with the force of gravity. If is known the Cm(t), is possible to deliver the right amount of current needed. The driver, according to the reference, adjusts the supply voltage of the motor, determining the current passage. Generally speaking, in the field of control, the unit control performs three basic functions:

1. Calculate the position reference θ_ref
2. Reads the current position θ
3. Calculates and generates the reference voltage for the driver Vref(t)

The unit control receives as feedback the position of the sensor and provides through the driver the reference voltage for the control. The cycle repeats until correct engine positioning. We define the position error as the difference between the reference position and the current position:

$$e(t) = \vartheta ref(t) - \vartheta(t) \quad [\text{rad}]$$

FIGURE 3.6: Position angular error.

To establish a useful action for the correct positioning is necessary to introduce the PID system of control. Proportional-integral derivative control, in short, PID control, is a feedback system widely used in automatic control systems. The PID control action is expressed as follows:

$$v_{ref}(t) = K_P e(t) + K_D \dot{e}(t) + K_I \int_0^t e(\tau) d\tau \qquad [V]$$

$$\frac{V_{ref}(s)}{E(s)} = K_P + K_D s + \frac{K_I}{s}$$

FIGURE 3.7: Elettric voltage for the automatic control system.

The

$$V_{ref}(t)$$

is the quantity that is given into the engine to correct the position and approach the correct one. The error that occurs with the engine is a quantity that can take both negative and positive values. When the target point is far from where we placed the tool we can say that the error is positive, while if we pass the target point we have to retract with the tool then the error will be negative.

The control voltage will be given by the sum of three rates: proportional rate, derived rate, and integrative rate. Within these elements appear "K" constants, known as earnings, which are named after each other. They are constants that must be selected for the specific application and field of action. These will be multiplied by the respective error.

- Kp will be calculated as Volt on radians
- Kd will be calculated as Volt on (radians) * second
- Ki will also be Volt on (radians) * second

PROPORTIONAL ACTION (READINESS)

The proportional action produces a command proportional to the position error Kp*e(t). This K is important because it takes into account the trajectory realized or followed, the objective is to reduce vibrations. It is the most intuitive and simple law to implement, as we have a direct proportionality between elements, the greater the error the greater the correction command. The proportion of the proportional action can be used alone, that is, a system can be created that "responds" actively to disturbances. It is said that the proportional share is an indication of readiness, as in the spring that responds almost instantly to external stresses to bring the balance back [Figure 3.8].

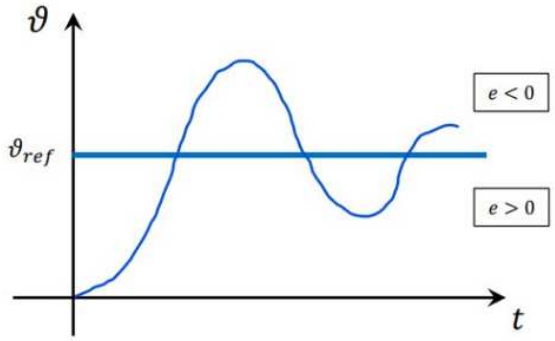


FIGURE 3.8: Proportional action to the position.

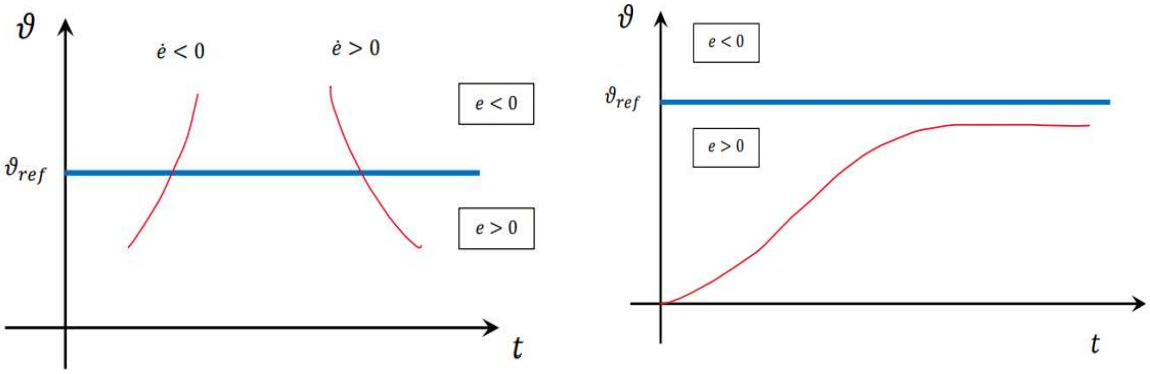
With this type of system planning instant step responses is impossible. Even by imposing an instantaneous step command, that is, by instantly checking for a very large error, the signal would be saturated. Moreover, by imposing a maximum controllable acceleration the engine will certainly go beyond the intended target, this is called overshoot, this system involves the "vibration" of the engine around a target point that prolongs the achievement and settling. When we go to the target you are without error and therefore there is no command (we are always late in the command). Eventually, it will come to a point where the system stops due to friction and the lack of torque required. The placement will not be exact and will also take place in a very long time.

DERIVATIVE ACTION (STABILITY)

To overcome the problems presented by a model consisting only of proportional shares, the derivative quota is also introduced. The derivative action produces a command proportional to the derivative of the position error Kd*ė(t). This K is a parameter that takes into account the presence of variations in the error in terms of increase or decrease [Figure 3.9(A)]. It provides an understanding of how our error is changing over time. If there is a derivative of the error or difference of position between the reference and control point, then we have the derivative term. The device can counteract high speeds of movement of the motors by limiting the speed of excursion during the movement and allowing you to adjust the latter to reach the target [Figure 3.9(B)].

- Counteracts/helps proportional action based on ė sign
- Anticipates overshoot, reduces readiness but introduces stability

It usually damps down to eliminate the overshoot, but be careful not to push too much in this damping otherwise, it will be impossible to reach the reference. In general, when we



(A) Direction of the effect of the derivative action. (B) Steady-state effect of the derivative action.

FIGURE 3.9: Derivative action to the position.

talk about a permanent state, the derivative action decays and the command action is only proportional ($\dot{e} = 0$).

INTEGRATIVE ACTION (ACCURACY)

To obtain greater control we can then introduce also the integrative action, this is often optional as the derivative action cannot be present alone in a control system. The integrative action produces a command proportional to the integral of the position error, that is, it takes into account the "history" of the error.

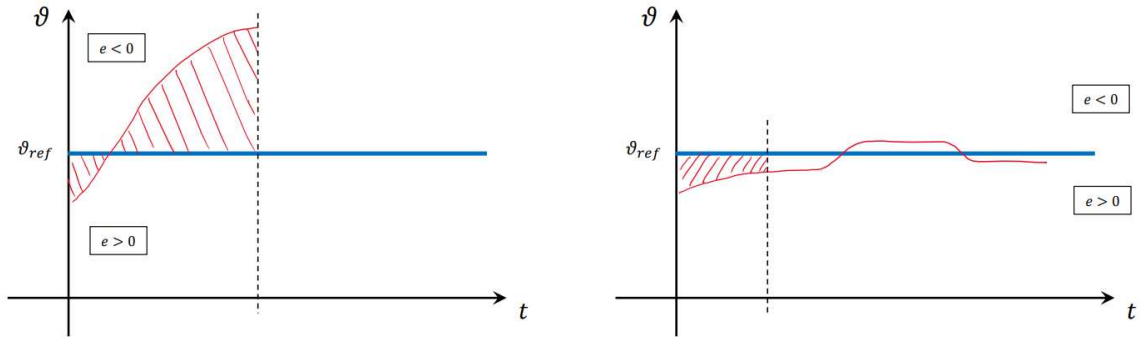


FIGURE 3.10: Integral action to the position, effect of the disease inducted by integral action.

This definition of the integral action causes the controller to have a memory of the past values of the error signal; in particular, the value of the integral action is not necessarily null if the error signal is null. This property gives the PID the ability to bring the process exactly to the required reference point, where only proportional action would be null. Since the action is always proportional to the integral, what happens is that the settling will be delayed

in time and vibrations occur around the reference, in fact will be an index of instability if the Ki is too large [Figure 3.10]. In general, however, this contribution adds accuracy to the system. The integration action introduces some difficulties:

1. Positioning effect of continuous oscillates, there will be always an area below the curves that isn't zero;

2. Fenomenum of "wind-up", unusual increment due to some error during the calculation of the area below

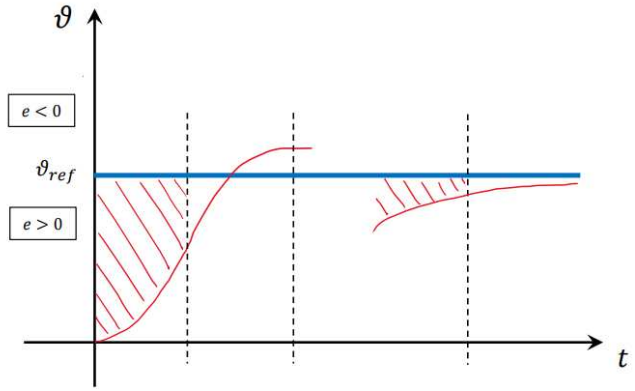


FIGURE 3.11: Steady-state effect of the integrative action.

We must be careful, the PID rates or constant can be positive or negative. Usually, a PID control is done by three positive rates is not a rule, the gains (Kp, Ki, Kd) can be positive or negative depending on the specific application and desired behavior.

- **Negative Proportional Gain (Kp):** A negative proportional gain can be used when the process variable and the control variable need to move in opposite directions. This is common in processes where an increase in the control variable causes a decrease in the process variable, or vice versa.

- **Negative Integral Gain (Ki):** A negative integral gain can be used to counter the effect of the integral term, which typically accumulates error over time. A negative integral gain can help reduce the steady-state error or prevent the system from winding up and becoming unstable.

- **Negative Derivative Gain (Kd):** A negative derivative gain can be used to dampen the system's response, effectively slowing down the rate of change in the control variable. This can be beneficial in systems where a rapid change in the control variable is undesirable or can lead to instability.

The choice of positive or negative PID gains depends on the specific requirements of the control system, the process dynamics, and the desired closed-loop behavior. It is essential to carefully tune the PID gains to achieve the desired performance and maintain system stability.

3.1.6 ROS system

ROS is an open-source, meta-operating system for robots. It provides the services you would expect from an operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management. Tools and libraries are available to obtain, build, write, and run code across multiple computers.

The primary goal of ROS is to support code reuse in robotics research and development. ROS is a distributed framework of processes (aka Nodes) that enables executables to be individually designed and loosely coupled at runtime. These processes can be grouped into Packages and Stacks, which can be easily shared and distributed. ROS also supports a federated system of code Repositories that enable collaboration to be distributed as well. This design, from the filesystem level to the community level, enables independent decisions about development and implementation, but all can be brought together with ROS infrastructure tools [3].

In ROS architecture software can be grouped into three categories:

- Tools for the development and publication of ROS-based software, known as ROS Community Level: ROS resources that enable separate communities to exchange software and knowledge.
- Libraries that can be used by client ROS processes such as roscpp, rospy, and roslisp, known as the Computation Graph: this is the peer-to-peer network of ROS processes that are processing data together. The basic Computation Graph concepts of ROS are nodes, Master, Parameter Server, messages, services, topics, and bags, all of which provide data to the Graph in different ways.
- Packages containing applications and code that use one or more client ROS process libraries, known as the filesystem level: concepts mainly cover ROS resources that you encounter on disk.

[3] Data from: http://www.ros.org

ROS is fundamental to extending the results and algorithms used to the widest range of existing robotic systems, expanding the applicability of such research. It's more than a simple environment of robotic systems development, it's an interconnected web of sharing for researchers and laboratories. Natively used in the industrial and automotive fields, the challenge is to extend this platform to wearable robotic systems built for humans. One of the strengths of ROS is its modularity and the possibility for different research groups to develop stand-alone components all relying on the same standard communication infrastructure.

3.2 Low-level Control

Low-level control in robotics refers to directly managing a robot's hardware components, such as motors, actuators, and sensors. It is the most fundamental layer of robot control, where algorithms and commands are translated into physical actions. Since it concerns the direct management of actuators and sensors, it ensures that the system responds accurately and timely to the commands. This type of control operates at a very high frequency and deals with operational details, such as force adjustment, velocity, and torque [Figure 3.12].

In the context of an exoskeleton, low-level control is crucial for several reasons:

1. Movement Accuracy: Exoskeletons must replicate human movements with great accuracy. Low-level control ensures that actuators move exactly as required by the system controls, minimizing errors and improving the accuracy of movements.

2. Response Time: An exoskeleton must respond quickly to user commands to be effective and safe. Low-level control handles commands in real-time, reducing latency time and improving device interactivity.

3. Safety: By directly managing sensors and actuators, low-level control can implement safety mechanisms to prevent sudden or dangerous movements that could cause harm to the user.

4. Energy Efficiency: Low-level control can optimize energy use, regulating the actuators' power efficiently and extending the autonomy of the exoskeleton.

5. Adaptability: Allows the system to adapt dynamically to changes in the environment or user behavior, ensuring smooth and continuous operation.

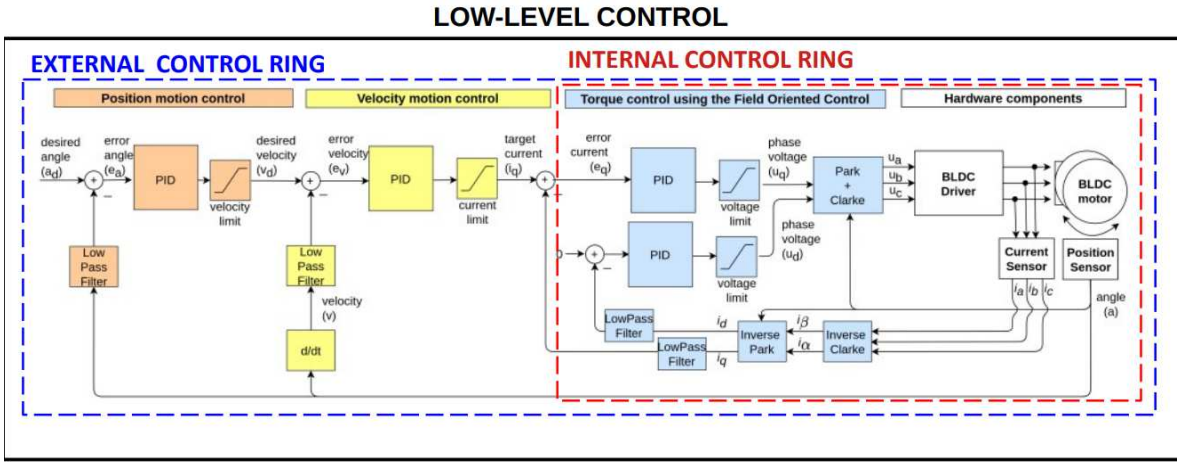


FIGURE 3.12: Low-level control scheme generated by a modular architecture for SimpleFoc. [4]

3.2.1 Internal control loop

This section aimed to program the motors' driver to generate the inner level of control, the torque, or current control. The SimpleFOC control technique was chosen since the motor is a high-performance brushless direct current. All the tests done are on the right knee joint because the low control is common for all the motors and actuated joints, so it was chosen to validate the system only with one joint and then apply for the remanent.

The internal ring consists of a feedback close control where the operator can introduce a current reference, known as target current "iq" [Figure 3.12]. The FOC system will elaborate this current, which converts the input into a voltage output thanks to a PID control. In turn, this voltage will be converted into three different motor pole quotes, thanks to Park and Clarke function. These transformations are fundamental to moving from stationary coordinate systems to rotating coordinate systems and vice versa, facilitating efficient motor control.

- Clarke's transformation converts three-phase currents (a, b, c) into a stationary two-dimensional (α, β) coordinate system.
- Park's transformation converts currents in the stationary $\alpha - \beta$ coordinate system into a rotating d-q coordinate system, aligned with the rotating magnetic field of the motor.

In the context of SimpleFOC, these transformations are used to convert measured currents in the motor phases into d-q currents and vice versa. This allows you to directly control

[4] Image Credit: $https://github.com/simplefoc/Arduino-FOC$

Q PID parameters for the torque inner control	
PID current q.P =	0.1f
PID current q.I =	1
PID current q.D =	0
PID current q.limit =	0.1f
PID current q.output_ramp =	1e3
PID current q.Tf =	0.1f

TABLE 3.1: Torque PID setting Q current parameters for inner control loop.

the current components that generate torque and magnetic flux, improving control efficiency. SimpleFOC implements these transformations efficiently, allowing precise and responsive control of BLDC motors. The feedback system is ensured by the Hall sensors. They read the magnetic field of the motor and convert it into a position output that will be soon converted again thanks to the inverse Park and Clarke function. At the end of the loop will be output as current "iq_new".

Thanks to the Park and Clarke transformations we have to operate only on the current "iq", and establish the entity of the limits control. To do this, the implemented code follows the SimpleFOC model, dividing the process into two voids.

The void setup, the first section of the code, is dedicated to the allocation of all the properties and limits of the elements that occur. Find the definition of the motor object, initialized with the number of magnetic poles of the chosen motor, the drive, the current sense, and the motor hall sensors. Follow some of the parameters, with the annexed module, that can be changed for the specific setup:

- Voltage Power Supply: 26 [V]
- Voltage Limits: 20 [V] Max DC voltage allowed
- Current Limits: 200 [Amps] - if phase resistance defined
- Velocity Limit: 20 rad/s (5 rad/s cca 50rpm)

The library offers various possibilities for motor control based on the readings we want to realize. Within the void setup, the Motor Type of Control must be defined, thus declaring the inner control ring.

For this type of application is essential to have some tools to filter and shield the information from interference or noise affected by the system. In this case for the internal control

D PID parameters for the torque inner control	
PID current d.P =	1
PID current d.I =	10
PID current d.D =	0
PID current d.limit =	0.1f
PID current d.output_ramp =	1e3
PID current d.Tf =	0.1f

TABLE 3.2: Torque PID setting D current parameters for inner control loop.

loop, the information is the current that needs to be filtered by a low pass filter. Such disturbances are mainly caused by the movement of the rotor, in which the permanent magnets are placed. These changes in electromagnetic flux, are a source of disturbances not only for the hall sensors placed inside the motor, but can also be for devices close to it.

Then, there is the void loop, which is the executive part of the code where the current ring of control is created and executed. We can choose which parameter want to plot, the input of the loop, and the tolerance of the two different low-pass filters.

As said before all the tests executed for the set-up of the inner loop control were carried out on the laboratory bench with a dedicated power system and clamped connection to ensure the safety of the operator and optimal control of the movements reducing the inertia. During the tests, we analyzed the torque output, the position output, and the velocity output of the motor with different current inputs such as step, sinusoidal, or cubic function. All of these tests were carried out to establish the PID parameters. The PID parameters for quadrature and direct current (being a vector control) are given [Table 3.1] [Table 3.2]. Note how the derivative term is not used since, through trial and error, we have seen that even with a small value, the control tends to become unstable. In addition, we have determined that as the derivative term increases, there is an increase in acoustic noise from the motor, at zero reference.

The choice of the parameters of the PID has been made on the type of application that the specific joint must carry out, the current and voltage limits that must always be checked, and also the quality of the torque response of the engine. Please note that the parameters are only functional for the specific motor coupled with the selected gearbox.

3.2.2 External control ring

In this section, I will talk about the position feedback theory with magnetic encoders, essential to the complete control code of the low level. As seen in the previous section, the inner loop can effectively manage torque and currents. This control is carried out employing the Hall probes. Hall probes are based on the Hall effect, the development of a transverse electric field in a solid material when carrying an electric current and placed in a magnetic field perpendicular to the current. Unfortunately, this type of feedback is functional only in the case of current torque. Hall probes are known to suffer from numerous disadvantages including:

- Magnetic field pollution: the magnetic field can be disturbed by external agents when the output limit is 20 mA or even lower.
- Extreme temperatures: places with large temperature variations can cause loss of accuracy.
- Read instability: continuous shows a band oscillation of the measured value.

The external loop control is based on the feed of angular position and velocity, the Hall probes aren't enable to read these two information without errors or oscillations. So as was suggested before, the system is equipped with magnetic encoders that can read this informations, the magnetic sensor being questioned is the AS5600 sensor presented in Chapter 2.

Hardware configuring

The AS5600, Figure 3.13, is also based on the Hall effect that uses planar sensors that convert the component of the magnetic field perpendicular to the chip surface into a voltage. The sensor consists of a small board inside which the chip is integrated. The board features 8 output pins for connection, 4 threaded holes for fixing, and a back tab that physically creates a short circuit between the 3.3V and 5V power supply routes of the chip.

1. VDD 5V Output for positive voltage supply in 5 V mode (requires 100 nf decoupling capacitor);
2. VDD3V3 Output for the positive voltage supply in 3,3 V mode (requires an external 1 micro-farad decoupling capacitor in 5 V mode);
3. OUT Analog/digital output essential to generate an analog/PWM output;

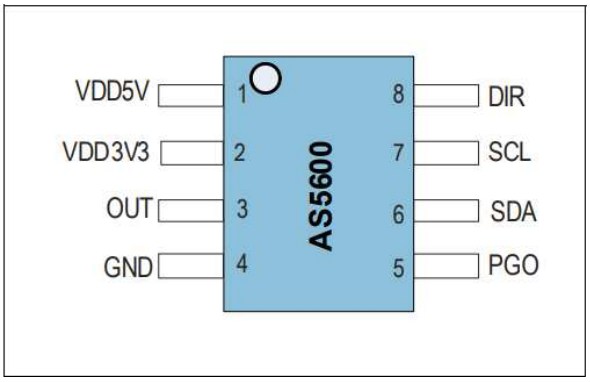


FIGURE 3.13: Representative image of the sensor.

4. GND Ground reference;
5. PGO Program Option;
6. SDA Channel for I²C data (consider external pull-up);
7. SCL Channel I²C for clock (consider external pull-up);
8. DIR Direction polarity for digital channels, an input pin (DIR) selects the polarity of the output relative to the direction of rotation.

If DIR is connected to GRD, the output value increases with clockwise rotation. If DIR is connected to VDD, the output value increases with counterclockwise rotation. In addition, to be able to change between the two power modes, the circuit must be adjusted by removing or adding a tab. The tab, supplied with the sensor, is nothing more than a bridge for the creation of a short circuit that binds the two power channels in one. The two possible power configurations are shown in Figure 3.14. In the case of 3.3V, there is a bridge, a short circuit, between the 5V and 3.3V inputs.

Operating conditions of power supply:

- VDD 5V Positive power supply voltage in 5.0V mode:
  - 4,5V Min
  - 5,0V Mean
  - 5.5 V Max value supported
- VDD 3.3V Positive power supply voltage in 3.3V mode:
  - 3.0V Min

- 3.3V Mean
- 3.6 V Max value supported

As can be deduced from the pin types on the board it is possible to connect the sensor through the communication channels SDC and SDA, which are essential for the communication of data in digital format, generally, this system is called I2C interface. I²C (short for Inter-Integrated Circuit) is a two-line serial communication system used between integrated circuits. The classic I²C bus consists of at least one master and one slave. The most frequent situation sees a single master and several slaves; however, multi-master and multi-slave architectures can be used in more complex systems.

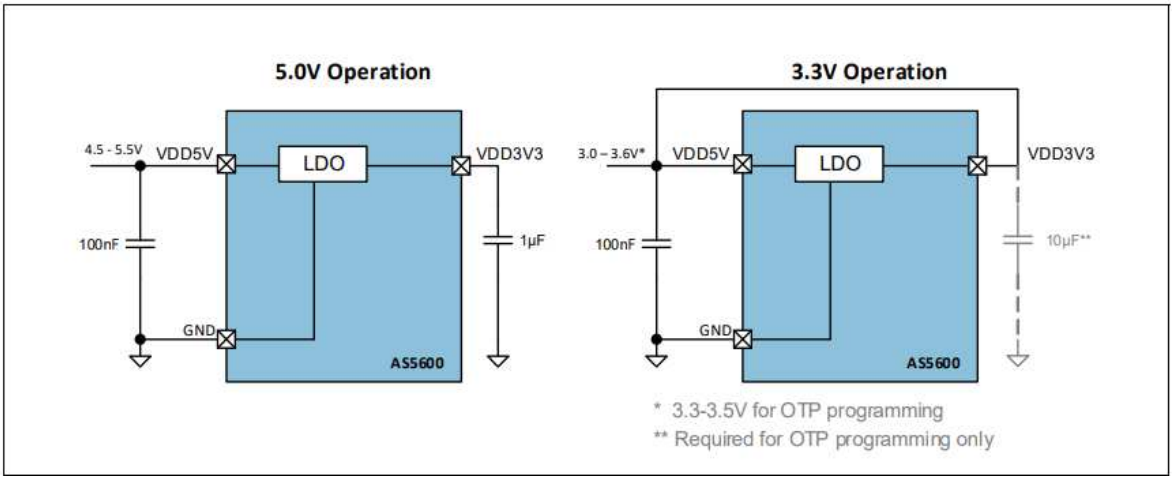


FIGURE 3.14: The two possible power configurations of sensor power supply.

It's also possible to communicate with analog output. This way is preferable because the raw information is faster than the converted digital one, and also because the measurement sensitivity is more elevated. By default, the AS5600 output stage is configured as an analog ratiometric output. The digital-to-analog converter (DAC) has a 12-bit resolution. In the default mode, the lower reference voltage for the DAC is GND, while the upper reference voltage is VDD. The output voltage on the OUT pin is ratiometric between GND and VDD.

The default range is 360 degrees, Figure 3.15. As shown in Figure 3.16, if the range is 360 degrees, to avoid breakpoints exactly at the limit of the range, a hysteresis of 10 LSB is applied. This hysteresis suppresses pin OUT switching when the magnet is close to zero or 360 degrees.

But doing so, in the default full-turn mode, the sensor returns an output that could, compared to a specific interval straddling a discontinuity, return values not accurate. Hysteresis

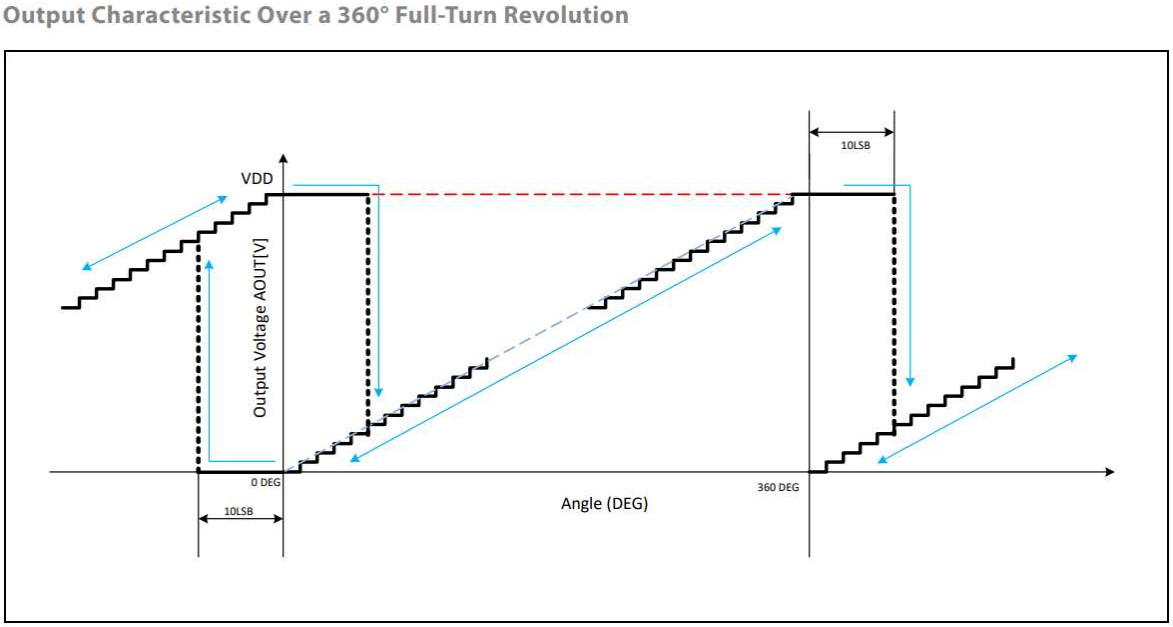


FIGURE 3.15: Output Characteristic Over a 360° Full-Turn Revolution.

shifts the angular value according to the difference of the bits. For a hysteresis of 10 LBS from datasheet the sensor processes from a difference of 2 bits. This could lead to a loss of information on intermediate values in an angular shift. If you advance beyond 360 degrees, then the signal on the OUT pin will remain unchanged for several additional "ticks" of the sensor (Vout = VDD), even if your tool keeps rotating. If you back up before 0 degrees, then the signal will remain unchanged for a certain number of "ticks" of the sensor (Vout = 0).

The number of ticks (clicks) you miss is exactly the hysteresis value you have set. In the standard configuration, this is 2 bits or four positions. Then the sensor will generate 4 more "ticks" before the value on the OUT pin changes again. By setting the sensor for a full turn with the PIN OUT setup, working at 5V we will have:

360/1024=0.351° Bit theorists.

1 bit = 0.351°

supposition then for 1° supposition = 2.844 theoretical bits.

With a simple calculation, we can see that 0.351° x 4 = 1.404° are the theoretical degrees that we do not take into account in reading. On a complete cycle, you will lose 8 ticks, corresponding to 2,844° missing. If a reduction in the maximum angular range is required, an operating mode is provided where the non-discontinuity points are pushed towards the edges of λ, where $\lambda = (360 - angle\_max)/2$.

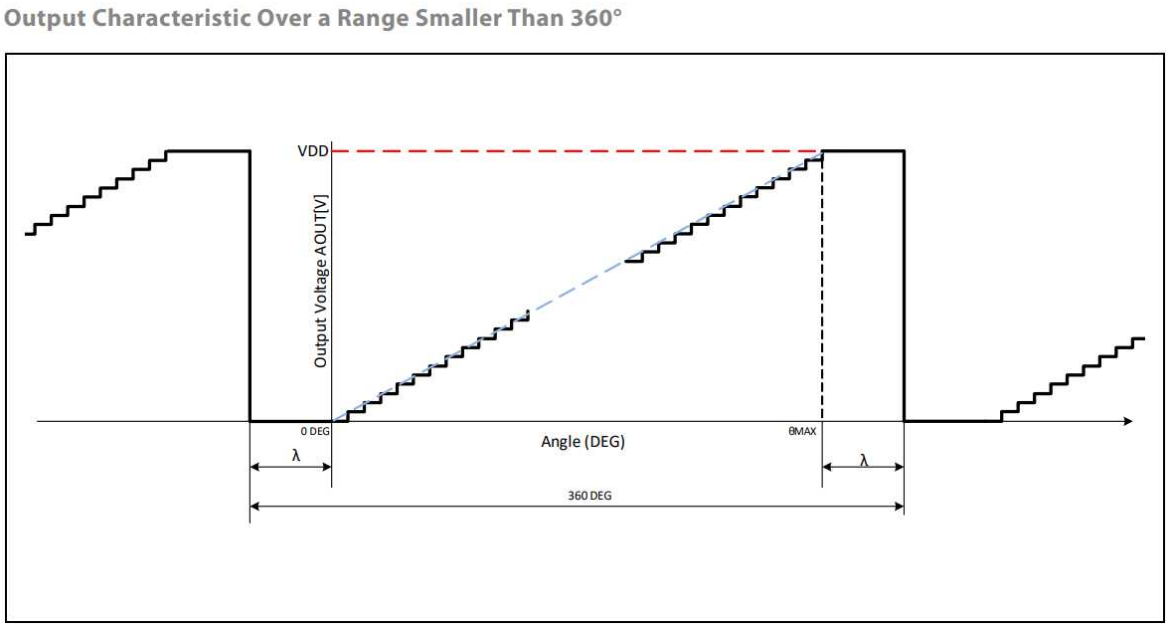


FIGURE 3.16: Output Characteristic Over a Range Smaller than 360°.

Implementation of the magnetic sensor

To ensure the correct reading of the sensor, we chose to implement a magnetic sensor with an auxiliary platform, the microcontroller Arduino Mega. The board can implement the connection with the encoder thanks to three cable connections, the ground, the voltage, and the analog output. We chose to use the sensor in the 5 V mode for the elevated accuracy and velocity of production. Using the software Arduino-IDE, software dedicated to the programming of microcontrollers, we could set the analog output for the angular reading, the direction of the turn, the maximum and minimum excursion angle, and also apply a filter to the parameter calculated.

The code implemented uses the C++ language, comprising a void setup and a void loop. Inside the void setup, there are the definition baud (115200), the set direction pin, and the set for output mode. While inside the void setup, there are all the operations necessary to receive a reading from the sensor. These operations are consequential:

1. Bit reading from the conversion of voltage: "analogRead(A0)"
2. Conversion from bit to a degree: "analog value * 0.35156"
3. Conversion from degree to radiant
4. Apply the offset during mapping: "mapToFraction(x, a, b, c, d)"

5. Calculate the velocity from the angle reading

6. Apply low-pass filter to the velocity for the noise shielding

The first product tests were carried out with a test device specially made for preliminary tests, and then on the bench right knee joint without joint feeding. The joint was moved manually and the readings were recorded via the plotter of the Arduino-IDE application. The first format of the code is the one proposed by the software.

Tuning of angular range

In this section, I will talk about the process of making the sensor useful for actuated joint applications. As said in the first chapter, all the native joints have a range of motion (ROM) that ensures the correct mobility without the possibility of injuries. In this application, we have built joints with a specific ROM, limiting the motion range mechanically. However, the sensor must return the correct angle correlated to the range that the specific joint can made [Figure 3.17].

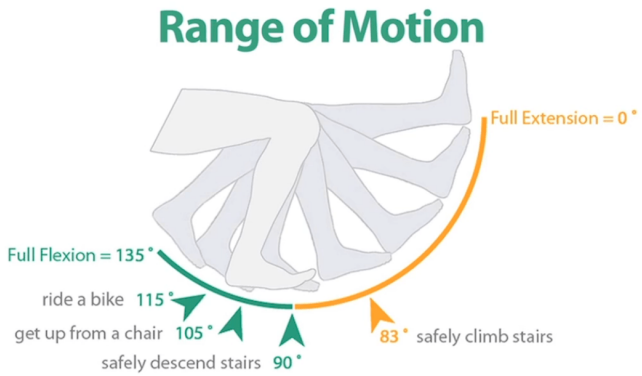


FIGURE 3.17: Range of motion of the right knee, full extension, and flexion [5].

To ensure these features we had to map the angular excursion of the sensor inside the running ROM, and this was done for each sensor. The first step consists of the knowledge of the ROM as an angular degree of the joint. To make an example I will talk about the right knee. The right knee has a range of motion where the angle is 0° for full extension while the full flexion is 135°. To generate the mapping we used the mapping function proposed by the IDE software, "mapToFraction(X, a, b, c, d)", where there are 5 parameters to introduce:

[5] Image Credit: $https://x10therapy.com/knee-range-of-motion/$

1. X: The angular read from the sensor
2. (a,b): The max and min excursion raw values
3. (c,d): The max and min excursion values mapped

The max and min excursion raw values are the values from the reading of the sensor without any type of mask. In the case of the knee, there are a minimum of 265° and a maximum of 357°. However, the max and min excursion values mapped are 0° and 93° respectively, we have reduced the ROM to increase safety during the movement. The X value is the output of the sensor converted in degrees and filtered from noise. The map function calibrates the scale from the raw angle to the new mapped angle, without losing sensibility or accuracy.

Attention must be paid to a single condition. If inside the raw max and min degrees angle is included the end-start of the scale 360°/0°, as showed before there is a fall in the value. This fall can generate problems of correspondence between the rag angle and the mapped angle. Moreover, there is the possibility of losing one degree of angular lecture. To avoid this type of situation is necessary to include in the code, before the mapping function, a "if" condition. If the raw value is higher than 180° the value will be recalculated as:

$$\text{degreeRAW} = \text{degreeRAW} - 360$$

With this strategy, we can avoid the fall. Below are the raw and new values used for the various joints

	Raw angle	Mupped values
Right hip*	132.19 to -0.35	-37.54 to 95.00
Right knee	265.00 to 357.00	0.00 to 93.00
Left hip	48.05 to 184.92	-37.35 to 95.19
Left knee	265.00 to 357.00	0.00 to 93.00

TABLE 3.3: Correspondence between the rag angle and the mapped angle.

Where the asterisk is present, the method described has been applied to avoid breakpoints.

The final process is to convert the implemented code into a ROS-compatible code. Inside the setup, there will be the initialization of the node and the definition of the message that will be published on the system. The sensor reading will be published on ROS as a message. The message will consist of a Data parameter that will include two values:

1. The angular value degree mapped

2. The velocity value, calculated as the simple derivate of the angular value and the previous one

The code is completely shown inside the A, B.

3.2.3 Set-up outer ring control

The steps necessary to tune a PID position controller with a cascaded structure are explained in this section. To completely define the low-level control, we had to work on the outer ring of control [Figure 3.18]. This control ring is composed of concentric sub-rings, the position ring, the external one, and the velocity ring, the internal one. They are linked by the information exchange. The input in the position ring is presented by the desired angle, while the input of the velocity ring is presented by the desired velocity. The desired angle is the precursor of the desired velocity, which is the precursor of the target current of the inner control loop.

The code and methodology used to design the two rings of the external control part are the same implemented for the internal control ring. But in this case, we will have two different sequences of control linked by some conversion and sensor feedback. We used the magnetic sensor to have feedback on position and velocity, the position feedback acts on the error angle, which is the input for the first PID control, while the velocity feedback acts on the error velocity combined with the desired velocity, obtained from the previous step, which is the input for the second PID control.

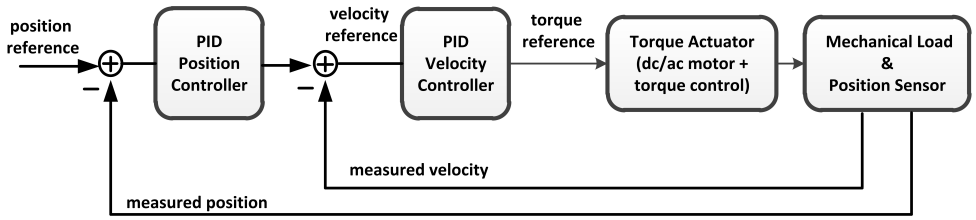


FIGURE 3.18: Outer ring control block scheme.

Two connected and sequential PID controls are difficult to set up contemporarily. The most recommended strategy to solve this situation is decoupling the control rings by working on the internal one and then acting on the second with the first process in action in the background. It's logical to concentrate on the inner loop for cascade structure adjustment since it's the circuit that's the fastest in practice. If all parameters of the outer position controller

are set to 0, the inner loop will be disconnected from the user position commands. This means working on the setting of PID parameters without the position feedback, silencing part of the code. The input, as velocity desired target, is included manually in the code.

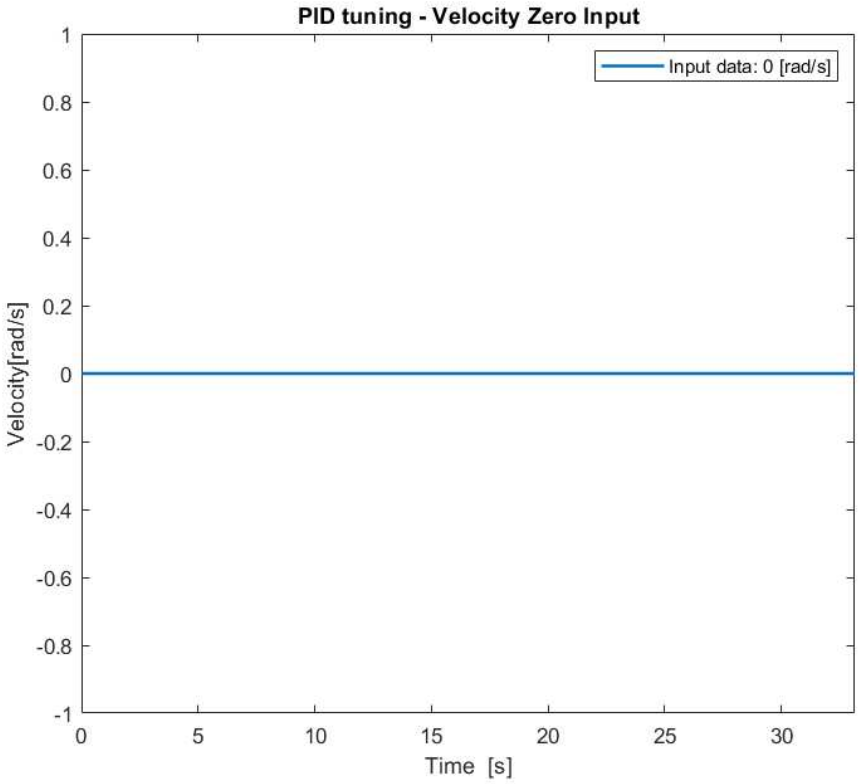
Velocity setup

To achieve a useful tuning we had to prove two different conditions: zero references and constant velocity referenced (positive and negative). In the first condition, the joint must rest in the starting position with a zero velocity output, while with a nonnull input, Figure 3.19(A), the joint must move to achieve the velocity we have done. The values of the PID constants are first assigned via the zero reference of the joint and then improved to verify a behavior consistent with the constant speed control.

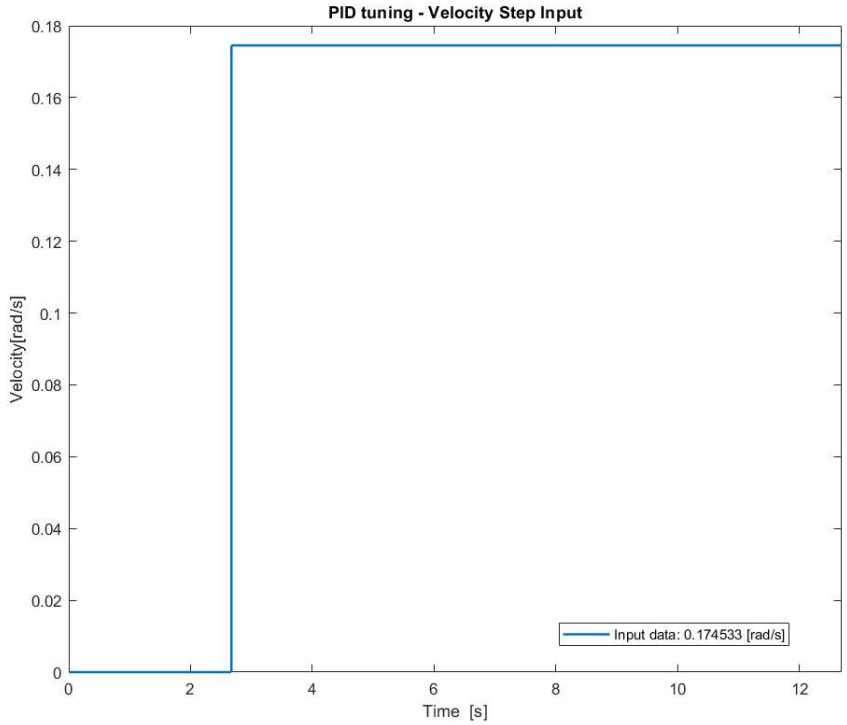
PID tuning step:

1. Starting with only the proportional Kp_V, with a value of 0.01 and the other K are 0.
2. Increase the value until the joint moves with cyclic sinusoidal movement around the expected value.
3. Increase the Ki_V, from a value of 0.01, until the joint movement is exasperated.
4. Increase the Kd_V, from a value of 0.01, until the joint movement stops and trembling at the expected value.
5. Increase (sharpen) the velocity controller as much as possible. As a rule of thumb, increase kp_V until you get close to the instability margin (at this margin, you will feel a vibration effect and some acoustic noise which is because of controller sharpness). Now, you can reduce kp_V to 90% of its value to increase the stability margin and remove vibration noise.

At the end of such a procedure, we expect that trying to move the joint manually will exert a lot of resistance, and the controller will quickly move the joint to the indicated reference, resting on the position where the zero speed is verified. Meanwhile, with a non-zero speed target, Figure 3.19(B), it will try to reach the speed input by increasing the current to keep it constant. The joint can be defined in a static state when it achieves zero velocity.



(A) Zero velocity input.



(B) Velocity step input.

FIGURE 3.19: Start condition of testing.

Position setup

The PID setup for the position control was initially done by iterative procedure following the list above. But in this case, the sensibility of the parameters was very high, that's because the two PID controls are linked together, and the two effects are summed. In this situation is impossible to find the correct parameters by a simple iterative procedure, it was necessary for a specific tool born for these tuned applications. The tool is included in the toolbox of Matlab by the name "PID tuner function[6]". The application needs as input three things:

- The system's Plant: it's the union of all the features that characterize the actual response of a system. It's represented usually by a transfer function.
- The output data, as an array, of the tests executed with the iterative parameters.
- The input defined data.

Unfortunately, the system was built excluding the effect that all the elements would have on the Plant system. For this reason, the Plant of the system is unknown. The first step consists of identifying the plant in the system through the application of the Matlab 'Plant identification' program. With the System Identification Toolbox software, we could use PID Tuner to estimate the parameters of a linear plant model based on the time domain response data measured by the system. Procedure for the identification:

1. **Loading of measured response data:** Load the measured response data for this example into the MATLAB workspace, When importing response data, PID Tuner assumes that the measured data represents a plant connected to the PID controller in a negative feedback loop. PID Tuner assumes that a step signal has been injected into the system input and that the system response has been measured.
2. **Import of response data for identification:** Enter the response data information. The output signal is the measured system response, output_y.
3. **Pre-processing of the data:** Depending on the quality and characteristics of the response data, it may be useful to subject the data to some pre-processing to improve the estimated plant results. PID Tuner provides several options for preprocessing response data, such as removing offsets, filtering, or extracting a subset of data.
4. **Adjustment of plant structure and parameters:** PID Tuner allows you to specify a system structure, such as One Pole, Underdamped Pair, or State-Space Model. In the

[6]https://it.mathworks.com/help/control/ref/pidtuner-app.html

Structure menu, choose the plant structure that best suits your response. You can also add a transport delay, a zero, or a supplement to the system. The plant structure and parameter values can be further adjusted to make the estimated system response more in line with the measured response data.

5. **System rescue and PID controller synchronization:** PID Tuner automatically designs a PI controller for Plant1 and displays a new closed-loop response in the Step Plot: Reference Tracking chart.

After that, to ensure PID's parameters calibration, we did the same two tests for the velocity, with a zero reference and a defined position. The PID setup is an iterative procedure, to ensure a perfect match between the input and output we have to repeat this type of procedure more than once; adjusting the parameters, and repeating again and again, until the system has a correct profile of the movement. There is one thing to pay attention to, during the PID tuning is possible to compromise the system's potential or develop instabilities. The operator has the work to find the compromise between accuracy, readiness, stability, and resonance vibrations. This compromise can be related to the particular assembled system or the devices used to communicate. We have considered these difficulties and operated to ensure the best response ever.

PID parameters for the velocity ring control in static tests
PID velocity.P = 2.25
PID velocity.I = 5.75
PID velocity.D = 0.25
PID velocity q.output_ramp = 1e3
PID velocity.limit = 12.00

TABLE 3.4: Torque PID setting velocity parameters for the inner ring of control in static tests.

PID parameters for the velocity ring control in static tests
PID angle.P = 0.4370
PID angle.I = 0.0000
PID angle.D = -0.0016
PID angle q.output_ramp = 1e6
PID angle.limit = 5.00

TABLE 3.5: Torque PID setting angle parameters for the outer ring of control in static tests.

Chapter 4

Next to control

4.1 Mid-level control

The term "mid-level control" (middle-level control) is used in automatic and robotic control contexts to describe an intermediate level of system management, which lies between high-level control (high-level control) and low-level control (low-level control)[Figure 4.1]. High-level control usually involves strategic planning, such as deciding the overall objectives of the system (for example, following a certain path, or reaching a specific position). The middle-level control takes these decisions and translates them into more particular commands that the system can execute. For example, it can determine the trajectories to be followed, the distribution of speed between components, or the management of interactions between different subsystems[Wills and Vachtsevanos, 2000].

In a robotic system, the middle-level control can manage the trajectories of movements, deciding how a robot must move to reach the target defined by the high-level control. For example, in a robotic leg, the mid-level control could determine the specific angles to be applied to each joint to perform a desired movement. The mid-level control can also supervise and adapt system dynamics, such as managing balance, avoiding collisions, or modifying behavior in response to changes in the environment or system state [Ali Nasr and McPhee, 2022].

Typical functions of middle-level control:

- Generation of trajectories.
- Coordination of the movements of several actuators.
- Adaptation of control parameters in real-time.
- Implementation of kinematic and dynamic models to predict the behavior of the system.

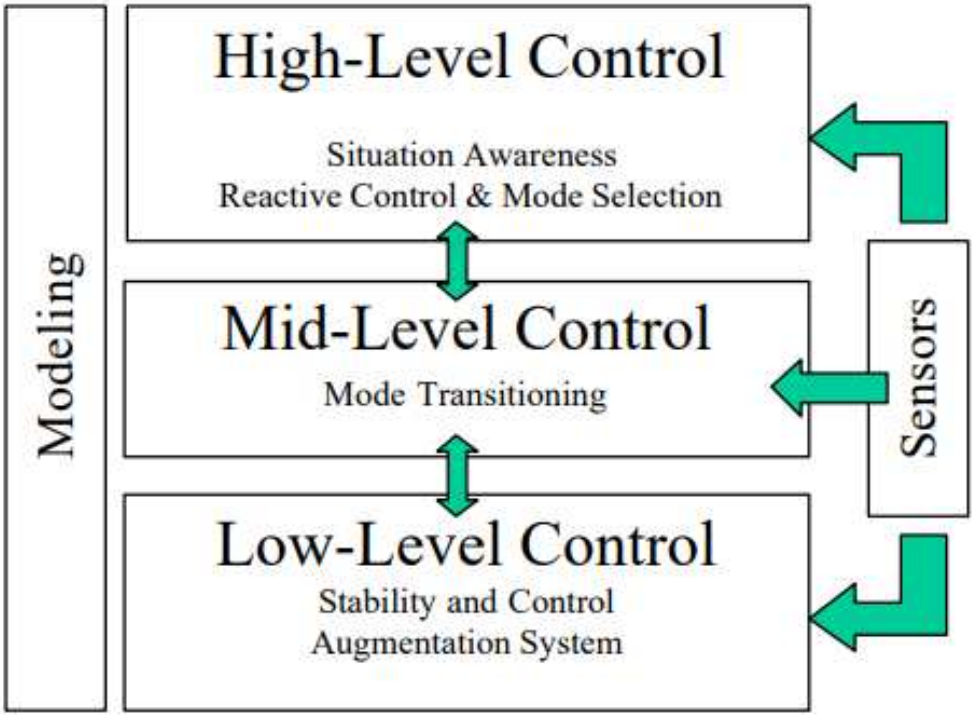


FIGURE 4.1: Block scheme of the control interconnected system [L. Wills, 2000].

4.1.1 Sinusoidal input

After accomplishing the low-level control the next step is to verify the ability of the system to reproduce simple curves with little lags and good accuracy. The named mid-level control consists of verifying the control system with a sinusoidal profile, of different entities, on the joint [Figure 4.2]. In this part is essential to adjust the PID control to correctly calibrate the effect of the joint.

The evaluation test consists of overlapping the two curves, the input and the output, observing the lag between the amplitude, the maximum and minimum value, and the mean value. Multiple tests consent to adjust the PID parameters and to achieve a mathematically correct profile. Of course, changing the inertia, the items, or the gearbox system all the iterative procedures have to be done again; it's important to establish a system model of control. The code implementing the sinusoidal input is included as a cyclical reference to observe the error gait development. The correct execution of this input is essential to ensure the ability of the system to reproduce complex commands. All the inputs tested are chosen to verify the system under the main condition of work and the voltage and current limits of

use. The code is completely shown inside the AppendixC.

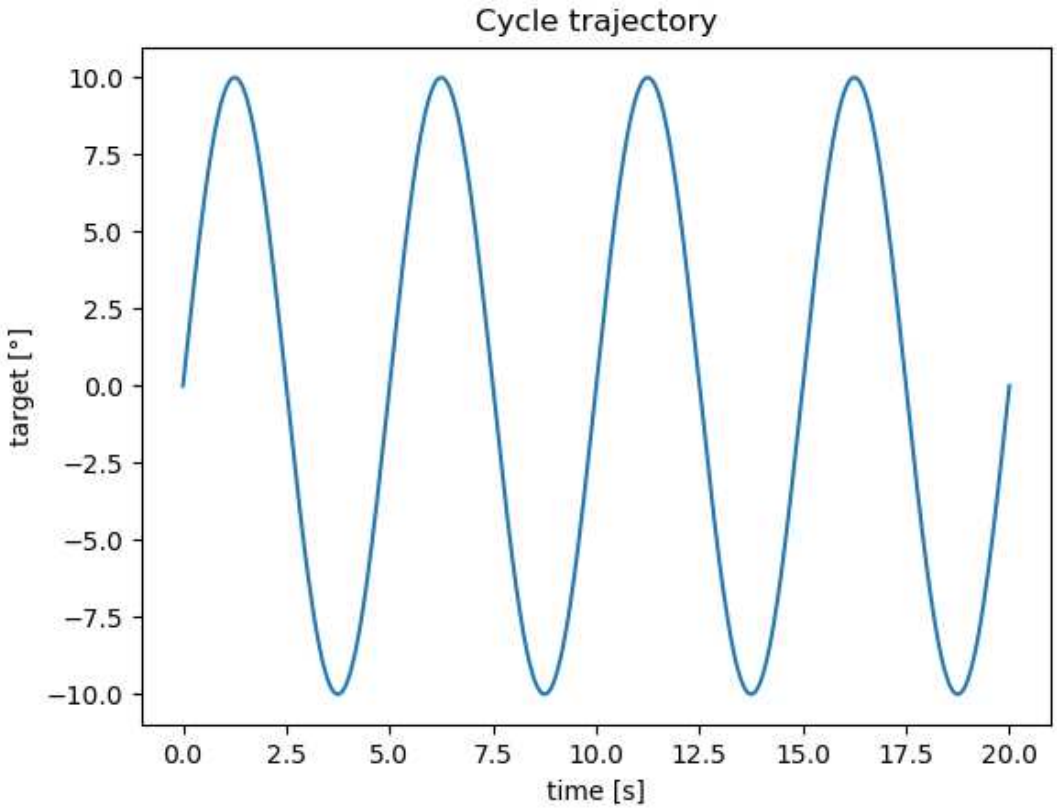


FIGURE 4.2: Sinusoidal input generated by our math law.

4.1.2 Gait profile

The fundamental part of this research is the qualitative and quantitative evaluation of the curves that the system can emulate. This section will be dedicated to the implementation of a code capable of sending complex position and speed references that repropose the physiological curves of the human step cycle.

It is essential to understand how using ROS can be done in the most direct and fast way possible by using a Python write path and the lunch files that will send the reference directly to the control system. The dedicated code section known as "gait trajectory generator" will present all the functions indispensable for the development of profiles on 4 different joints. Again, a single joint will be used as a reference to assess the effectiveness of the strategy and the elements implemented.

PID parameters for the velocity ring control in dynamic tests
PID velocity.P = **2.25**
PID velocity.I = **5.75**
PID velocity.D = **0.25**
PID velocity output_ramp = **1e3**
PID velocity.limit = **12.00**

TABLE 4.1: Torque PID setting velocity parameters for the inner ring of control in dynamic tests.

PID parameters for the velocity ring control in dynamic tests
PID angle.P = **2.874**
PID angle.I = **0.015**
PID angle.D = **-0.25**
PID angle output_ramp = **1e6**
PID angle.limit = **5.00**

TABLE 4.2: Torque PID setting angle parameters for the outer ring of control in dynamic tests.

Code steps implemented:

1. Definition of cycle duration, frequency, amplitude, presence of offset, and the constant to create the curve.
2. Compiling the function for the position angle profile of the hip and knee. The functions consist of a sum of sinusoid and cosine scaled with the constant factors and period.
3. Compiling the function for the velocity profile derivating the precedent curves
4. Create the plot preview of the motion control
5. Run the joint

The choice of constants and the sequence used to compile the profile angle curves are selected from the literature to maximize the similitude between the curve created and the human profile[Figure 4.3][Figure 4.4]. Also in this case the goals of the tests are to prove the ability of the system, the readiness, the accuracy, and most importantly the stability during the movements. These are dynamic tests done with all the safety systems on, and with a workplace free from objects and obstacles.

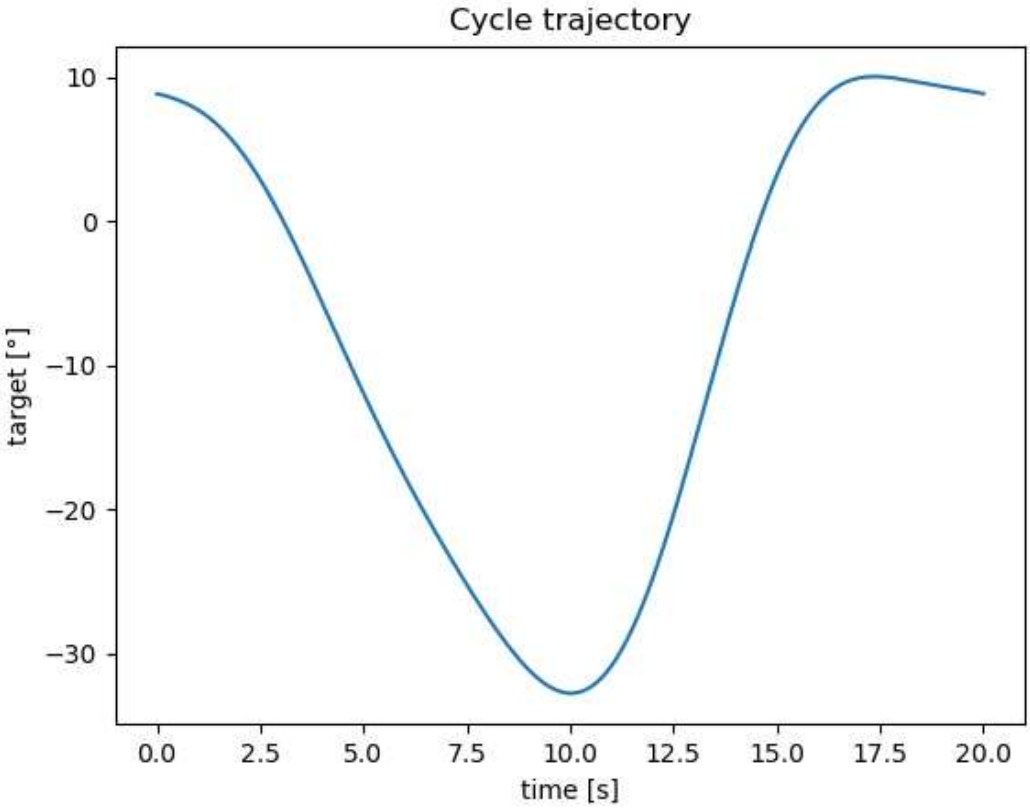


FIGURE 4.3: Gait cycle for the hip right joint generated by our math law.

4.2 High-level Control

While the low-level control system works through the execution of point commands and therefore the control is based on the comparison between the input and the output the high-level control is based on the execution of complex tasks that the system must perform. It must constantly monitor to correct errors or to introduce changes to the command itself. The high-level system is therefore fundamental to develop an exoskeletal device capable of correctly performing position profiles consistent with the native movement of the human body. It is therefore obvious how the optimization of the high-level system will lead to optimal compatibility with the human motor system especially when the system is on an electric scale and is limited in terms of degrees of freedom compared to the human body. A system with fewer degrees of freedom will have fewer possibilities in terms of movement and therefore adaptation.

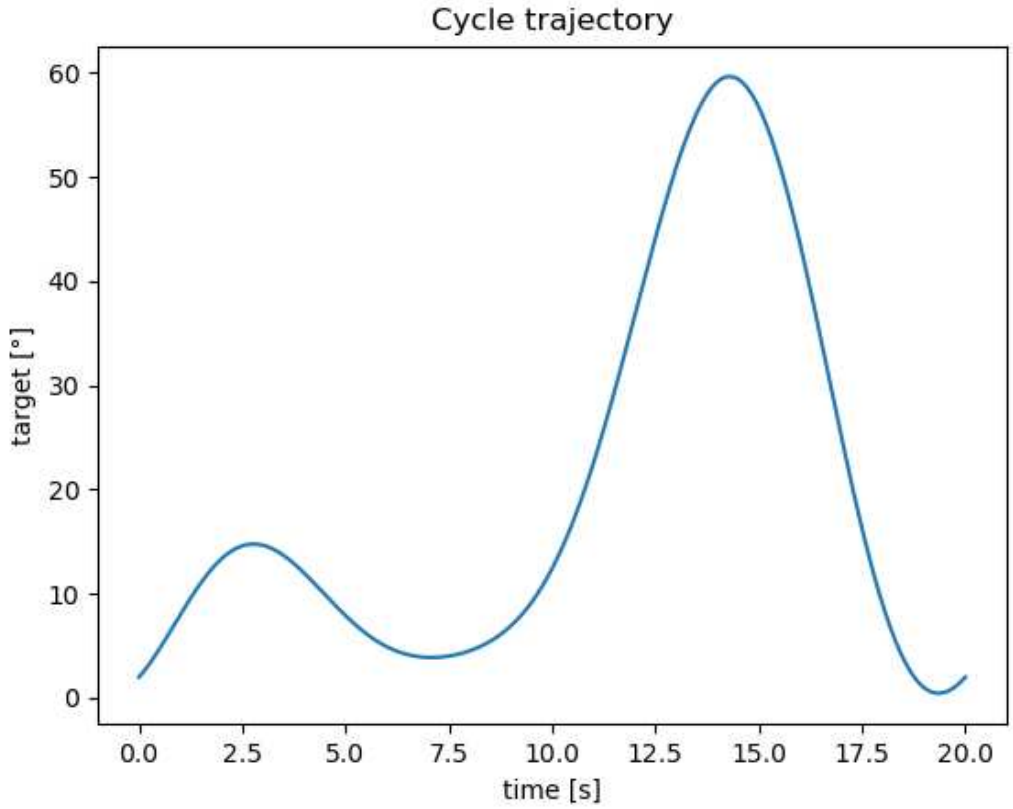


FIGURE 4.4: Gait cycle for the knee right joint generated by our math law.

Components of High-Level Control

- Planning of the Movement: Defines the sequence of movements needed to achieve a specific goal. Use planning algorithms to determine the optimal path avoiding obstacles and ensuring the safety and effectiveness of movement.
- Generation of the Trajectories: It converts the motion planes into time trajectories that the system must follow. Trajectories are described in terms of position, speed, and acceleration over time.
- Control of the Stability: Ensures that the system remains stable during operation. In exoskeletons, this may include maintaining user balance and preventing falls.
- Human-Machine Interaction: It manages how the system interacts with the user. This may include interpreting user commands, adapting to the user's natural movements, and providing sensory feedback.

- Adaptation and Learning: Includes machine learning techniques and online adaptation to improve system performance over time. The system can adapt to changes in operating conditions or user capabilities.

In particular, the high-level supervision system integrates a continuous cycle of planning and execution of the route that can react quickly to environmental events and human interventions. The contribution of the ROS platform is fundamental in this context, without which control of such kind would be unthinkable and excessively complex in terms of architecture and computational burden. Particularly in this field, we talk about two types of human-machine control:

- The direct control, with some controller or joy-con link to the EXO
- The neuro-references to control, the users achieve the control with their body signals like EEG and EEM.

The first type acts in the initial phase of the project where the system has to be debugged and set, it's also used to do easy tests with users during the evaluation. With direct teleoperation, the user has complete control over the robot through the HMI(Human-Machine-Interface). Commands are directly mapped to robot actions with no need for processing. In this sense is possible to use some types of artificial intelligence to mediate between the device and the users. The most common strategy is shared control, shared control is effective in several applications that require precise operations, it also improves the independence of people with disabilities in both daily living mobility and manipulation [Tortora and Gottardi, 2022]. These solutions are often not optimized and efficient with the consequence of an increased risk of technical faults

The second one is the next step to achieve complete control and a system autonomous, that can be used for unhealthy people with severe motion conditions. In this case, we talk about ROS-Neuro, ROS-Neuro was created to be the initial open-source neurorobotic middleware that puts human neural interfaces and robotic systems on the same level of understanding and implementation. ROS-Neuro provides several standard interfaces for acquiring neurophysiological signals from different commercial devices, processing EEG and EMG signals using traditional methods, and classifying data with common machine learning algorithms. With that, we can implement a bridge between the intention of the users to do an action and the execution of the device. Overcoming obstacles, performing complex movements, or modifying an action will be possible through neural communication and robotic artificial intelligence; which will mediate information, making it understandable for the system and converting it into actions [Tortora and Menegatti, 2022].

ROS as a platform for the development and control of robotic systems involves the implementation of both communication models with the user. In this design phase, the code developed only allows for a direct interface to be included with the user. To achieve complete neuro-control are also essential the presence of cameras or other environment mapping gadgets to obtain, from the external space, input to recalculate the command of control for safety movements.

Chapter 5

Tests and Results

5.1 Encoder test

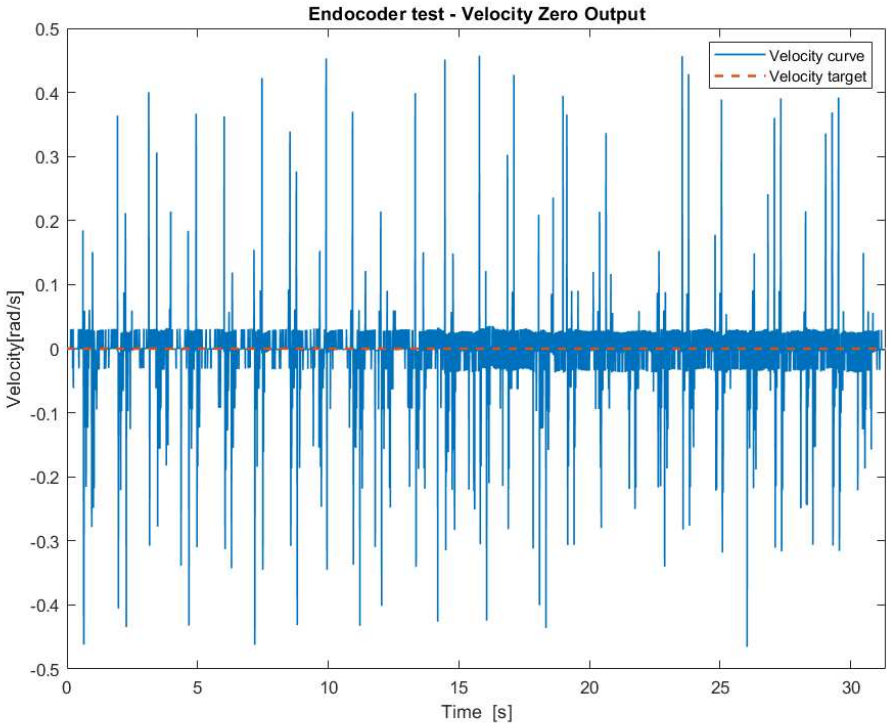
The first property that we verified is the sensor's quality of the signal using the serial plotter of ArduinoIDE. The plotter consents to show all the values, from the bit value decoded to the velocity calculated by the derivative. The signals that are interesting for us are the mapped degree angle signal and velocity. We reproduce these curves into the Matlab plotter for a better view. Initially, only the velocity was filtered, so the angular degree was a pure value.

Unfortunately, the position value was also affected by some spike signal and noise from the magnetic interferences, and the velocity signal was still too noisy [Figure 5.1]. The result was not unsatisfactory, but not useful for the intended purpose. We don't desire irregular movements due to casual errors. So, we decided to introduce another filter for the position and increase the effect of the velocity filter.

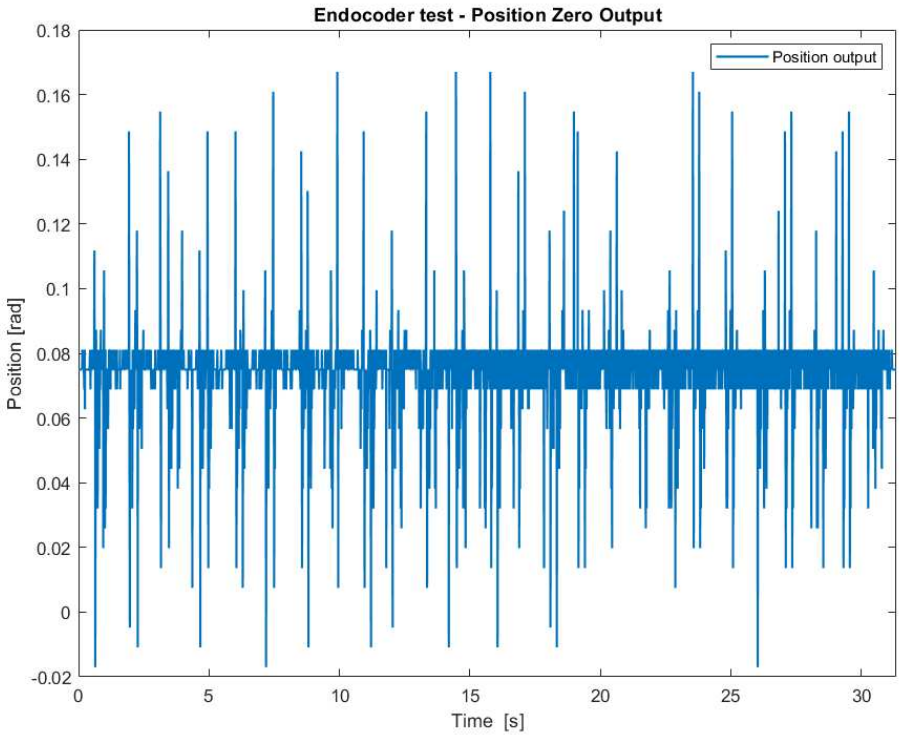
It's clear that when an operator uses a low pass filter it has to verify two conditions:

- The definition of the signal, absence of noise or errors;
- The velocity of reading, the readiness during dynamic changes.

A higher value of shrinking ensures an excellent definition of the signal, but a higher lag between input and output. While a lower value of shrinking ensures readiness, but a noise in the background. After various tests, we decided to elect a system with a higher readiness rather than the noise in the background. In the end, all two tests showed excellent results and a signal with a tiny type of noise below the threshold of interference with motion [Figure 5.2].

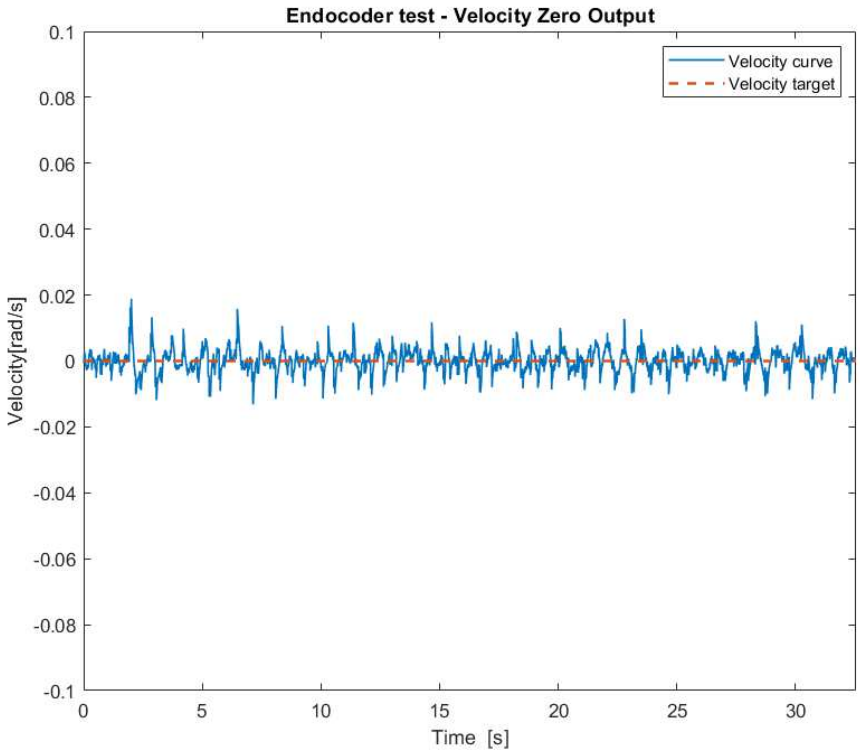


(A) Velocity data plot without filter.

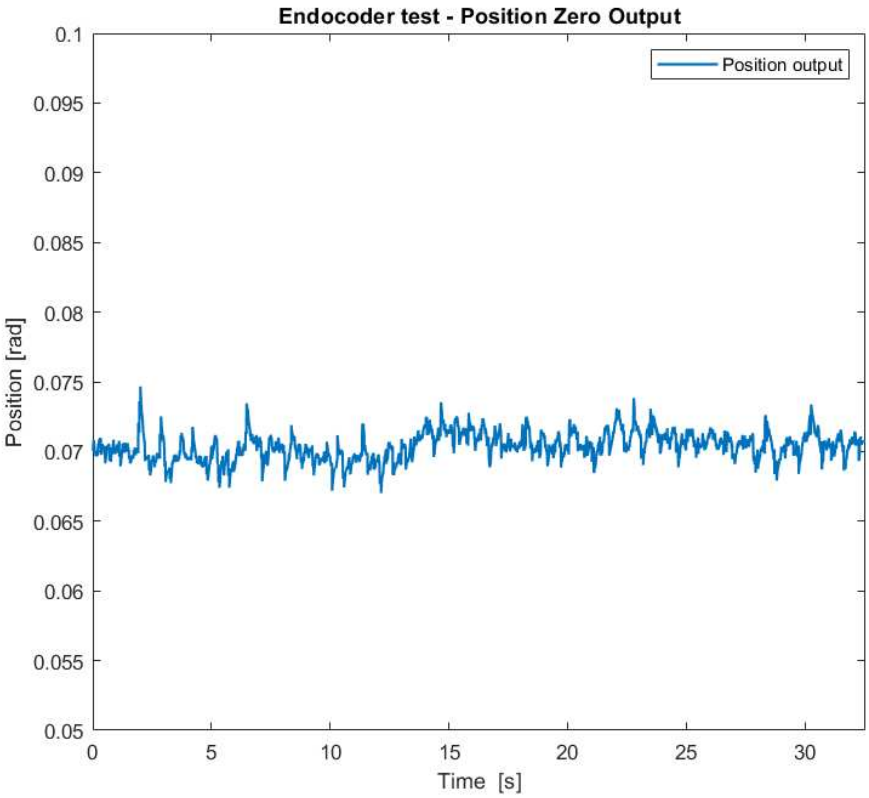


(B) Position data plot without filter.

FIGURE 5.1: Raw signal from magnetic encoders.



(A) Velocity data plot with low pass filter.



(B) Position data plot with low pass filter.

FIGURE 5.2: Filter signal from the magnetic encoder.

The sensibility of the reading is optimal, and the velocity, with an accurate filter tuning, shows a good signal.

5.2 Static control test

This section discusses the tests done on the right hip joint to achieve accurate and ready position control. The hip was selected because it represents the joint that will bear the most inertial loads. Also, the control on the hip is more complex given the wide range of angles it can achieve, indicating that an optimal hip setting would be effective, without small adjustments, even for the knee. The results with the knee are presented in the AppendixD.

FIGURE 5.3: Image of the test-box system.

Figure 5.3 represents our test-box system to prove the functionality of the hip joint on the right leg. We tested only one part of the complete system because the tuning procedure was the same for all the two parts.

5.2.1 Rest input control

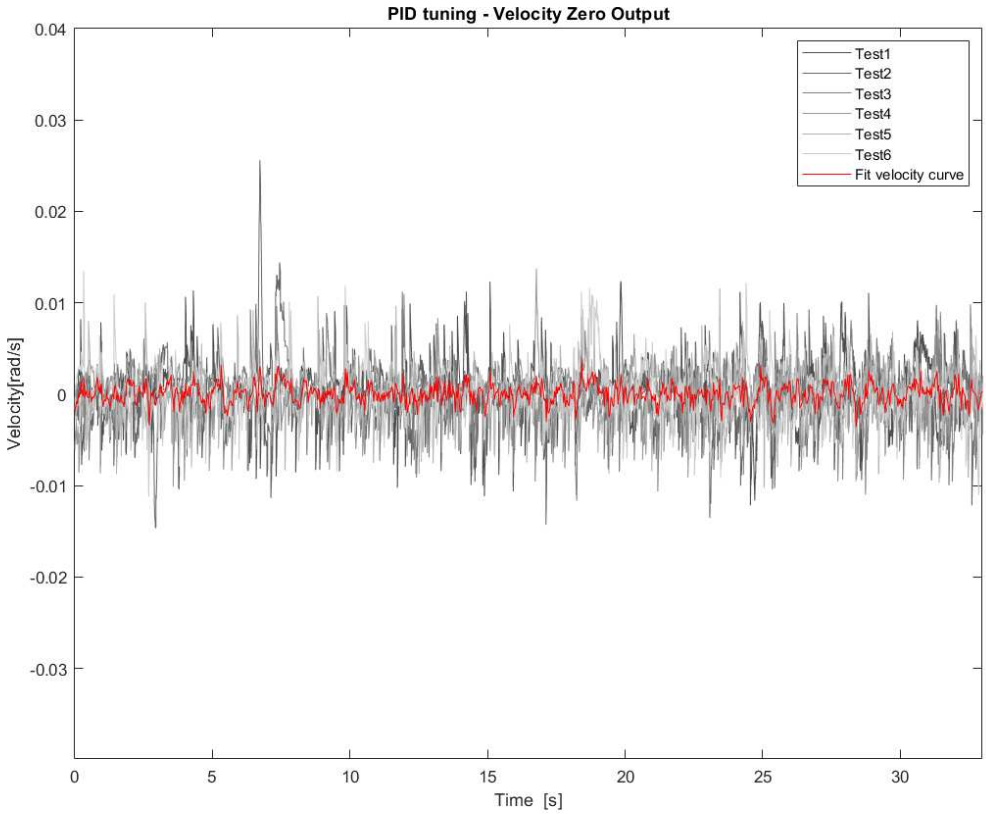
Velocity control ring enabled

The first condition proved was the static condition with a flat steady state with the velocity control ring enabled. The answer we hoped to obtain was a flat signal in velocity and the absence of an angular spike in the position signal. The steady state is essential to achieve effective control during the rest of the walking phases or to ensure the ability of the system to rise a step. A zero input represents the input of the test. More than one test was done, clustered into six groups, and synthesized in an average function with a special command known as 'smoothing spline'. The position that we tested is the perfect vertical one, with a preventive set-up of the joint to the position equal to zero.

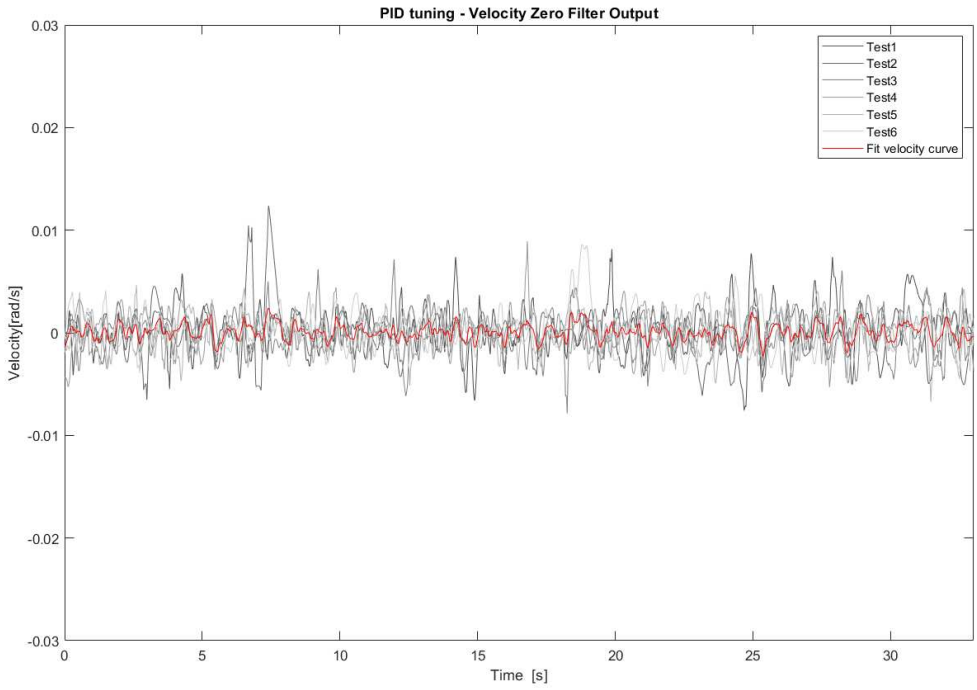
A grayscale represents the test curves, while the red curve is the fit-smoothing spline from the six tests, it represents the mean trend of all six tests. The first effect that can be seen is the noise and the absence of a perfect flat signal of velocity, the joint moves around the steady state value, as can be seen in Figure 5.4(a) and Figure 5.5(a). The maximum variation from the velocity target is less than 0.03 [rad/s]. This means that the joint is moving around the target with a nonnull velocity of less than 1,718873 [deg/s]. The actual response is certainly affected by noise, and the real movement of the joint is hidden. To understand the effect of the error we have to remove the noise in the background with a smoothing function, acting with a moving mean and a robust linear regression.

Removing the mean effect of the noise's error makes it possible to see the actual movements of the joint during a zero stay command, Figure 5.4(b) and Figure 5.5(b). In this case, the maximum variation from the velocity target becomes less than 0.015 [rad/s]. This means that the joint is moving around the target with a nonnull velocity of less than 0,85943669 [deg/s]. These types of oscillations are spikes of movement of the motor, too tiny, that can't act on the joint but create a hardly noticeable vibration to the user. Although the response is not flat, it is a sufficiently stable signal.

To complete the analysis we have to observe the position output of the joint. Where can appreciate the effects of the velocity unstable movement. The curves show the signal just filtered with the same procedure discussed before. The result is obtained without the presence of the position control able, but only with the velocity ring control [Figure 5.6]. Before the velocity spike of some curves, the position mean is near to the position that typically the system has to follow, but there is evidence that after the little spike, the position has a click, creating an error of 0.0071 [rad] equal to 0,4068 [deg].

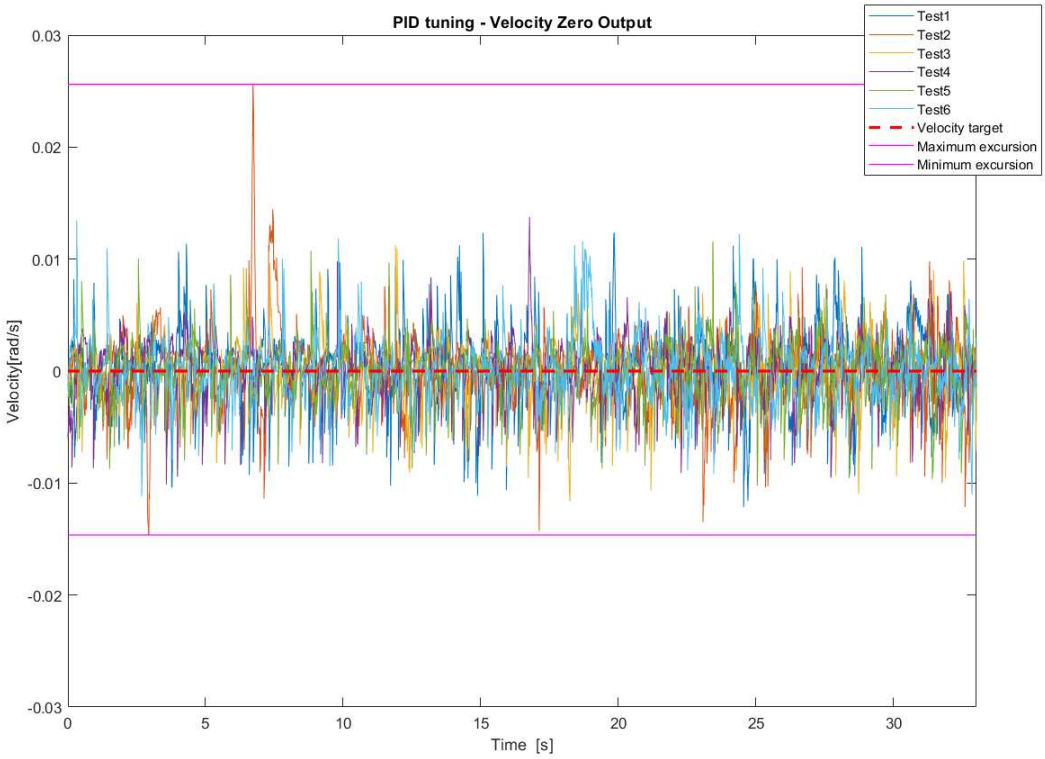


(A) Red mean signal of the velocity output.

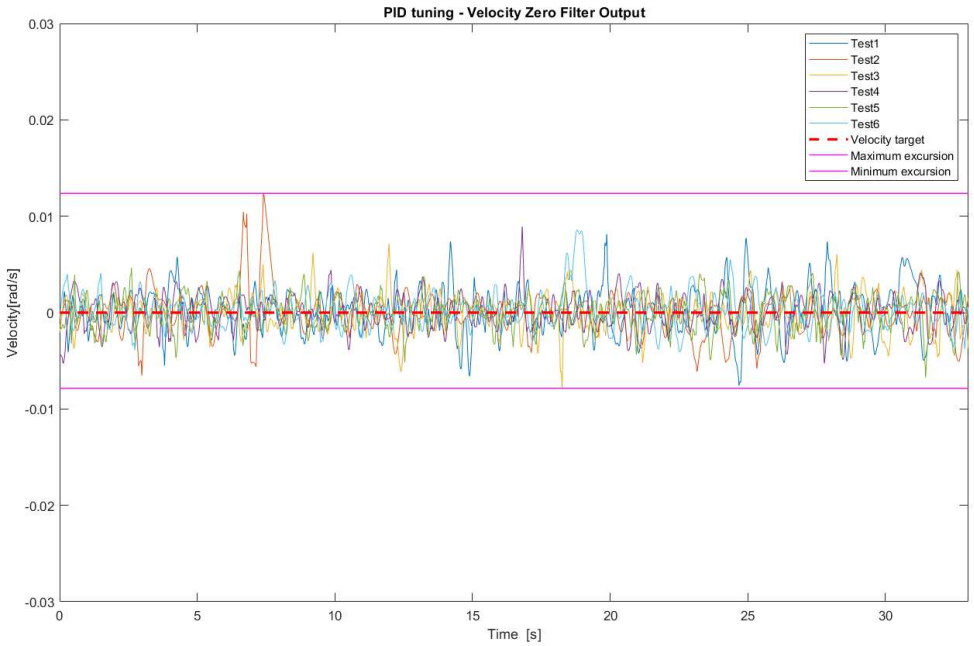


(B) Red filtered mean signal of the velocity output.

FIGURE 5.4: Example of velocity output for zero velocity input, grayscale and mean.

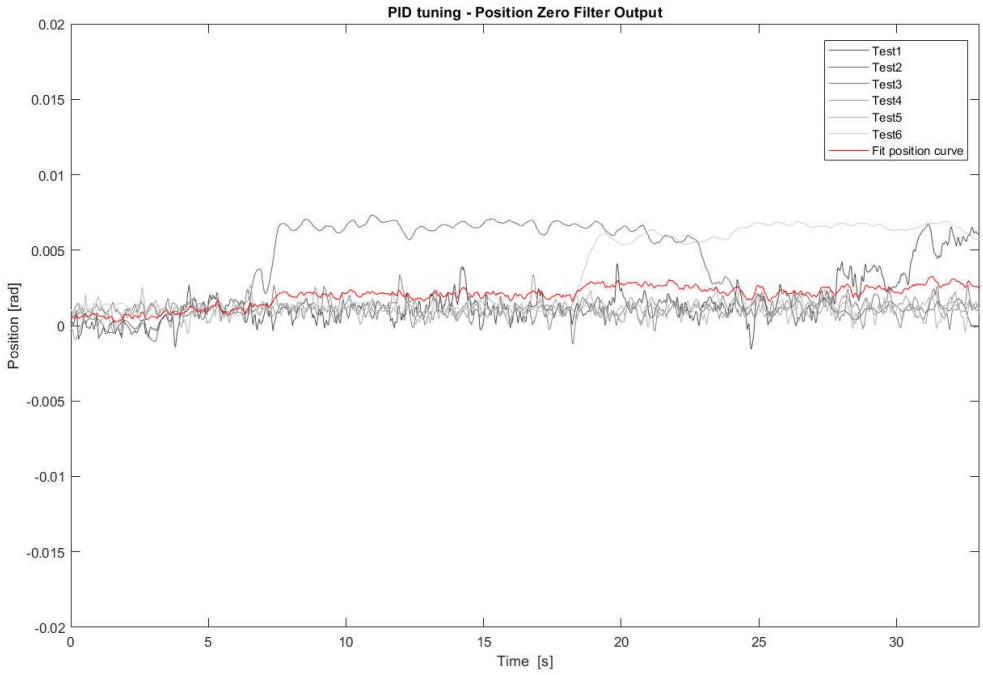


(A) Signal of the velocity output with the excursion.

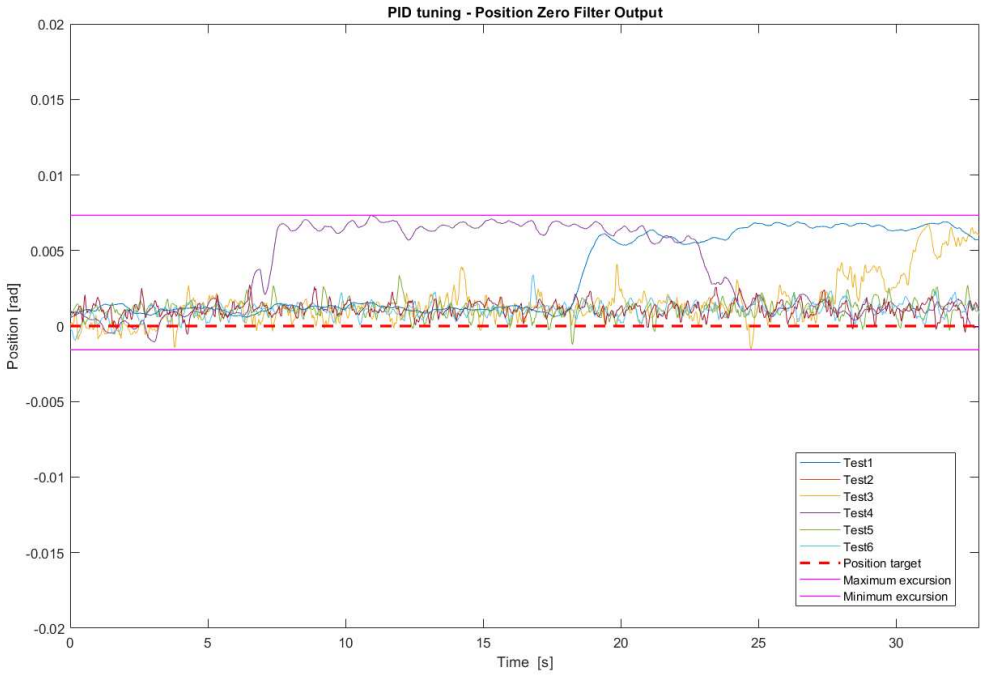


(B) Signal of the filtered velocity output with the excursion.

FIGURE 5.5: Example of velocity output for velocity zero input, color scale, and delta.



(A) Position output with mean, greyscale.



(B) Color position output with mean and the excursion.

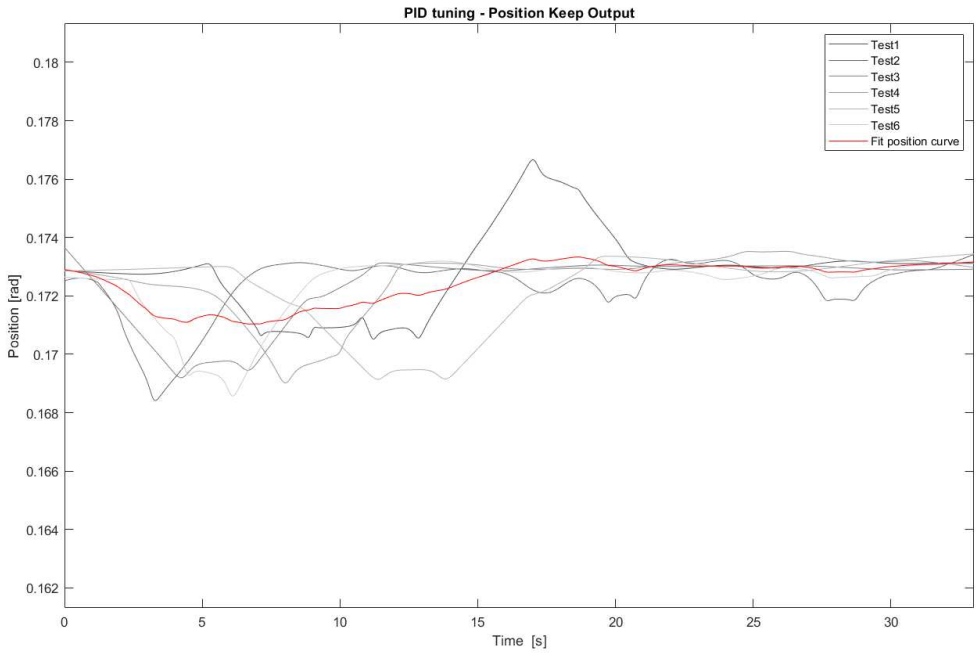
FIGURE 5.6: Example of position output for zero velocity input.

This is the principal reason for a possible hardly noticeable vibration to the user. After that looking at Figure 5.6(b) is possible to see the presence of a systematic error that imposes a deviation from the target desired. It is important to understand that the position control ring was disabled so the joint was able to adjust the position to maintain the velocity output near the velocity target. Meanwhile, the position clicks are under 1/10 deg, it is evident the need for control in position at the same time as the speed.

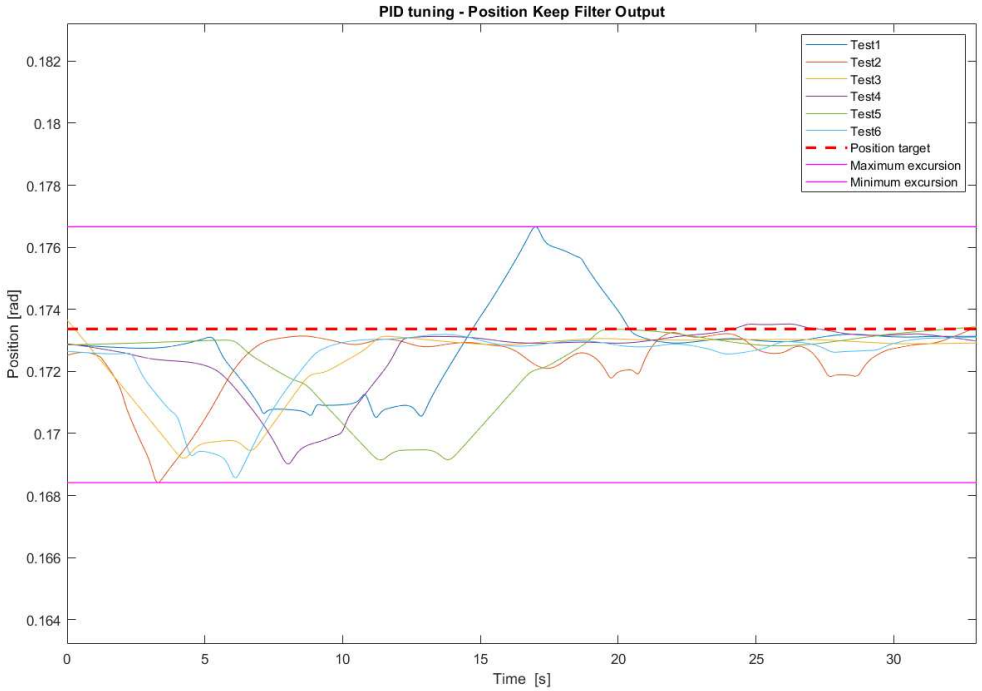
Position control ring enabled

Enabling to activate the controls simultaneously it can understand the real effect of the control. The two control rings collaborate in one system to ensure a better condition of control. It can be seen how the mean tendency is settled to the real value of the position imposed, Figure 5.7, but unfortunately, the control system fully able with all the control rings is not enough strong to avoid the tiny spikes click born from error and noise. The system tends to move for a reposition operation cyclically in time when the story of the error is big enough; this is the effect of the Ki PID operators. Looking the Figure 5.7(b) is possible to understand the entity of the reposition. The movements are less than 0.06 [rad], which means a movement less than 0.3 [deg].

It is important to note that the angular value shift is always contained within a range of motion less than one-half degree. This result is acceptable overall since the only moment of real system stasis for the application required is when the legs are fully extended for lying in the standing position. In the presence of a load such as the body system will not be able to affect any type of clicks, leaving on the contrary a certain value of compliance to the user. Furthermore, the presence of this error click can be due also to the experimental set-up desk, where the vibration of the platform can be the trigger to the click reposition.



(A) Position output with mean, greyscale.



(B) Color position output with mean and the excursion.

FIGURE 5.7: Example of position output for zero position input, grayscale and mean.

5.2.2 Step input control

Velocity control ring enabled

Following the test, the next step consists of a non-null value of input. We have introduced a step command to the velocity control, evaluating the velocity step response with the higher step value for this type of application, 10 [deg/s] equal to 0.174533 [rad/s]. Analyzing Figure 5.8 we can see the presence of two systematic errors:

- Starting overshot about 0.3/0.6 [rad/s]
- Discrepancy between target and plateau result

The overshoot depends on the proportional term which acts more slowly than the change in inertia. The joint has to move from a point with different inertia contributions that act on the initial current input. We can reduce this effect by operating on the Kd of the velocity PID.

Also in this case we can remove the noise on the background of the signal highlighting the real slope of the joint motion. In this field, the code imposed to filter the signal was the simple moving mean with a smoothing factor of 0.05.

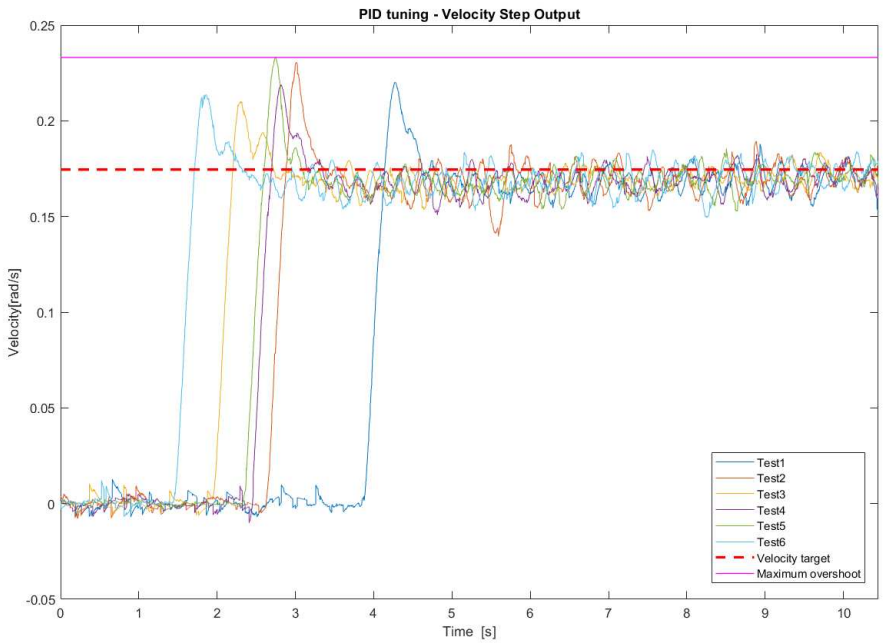
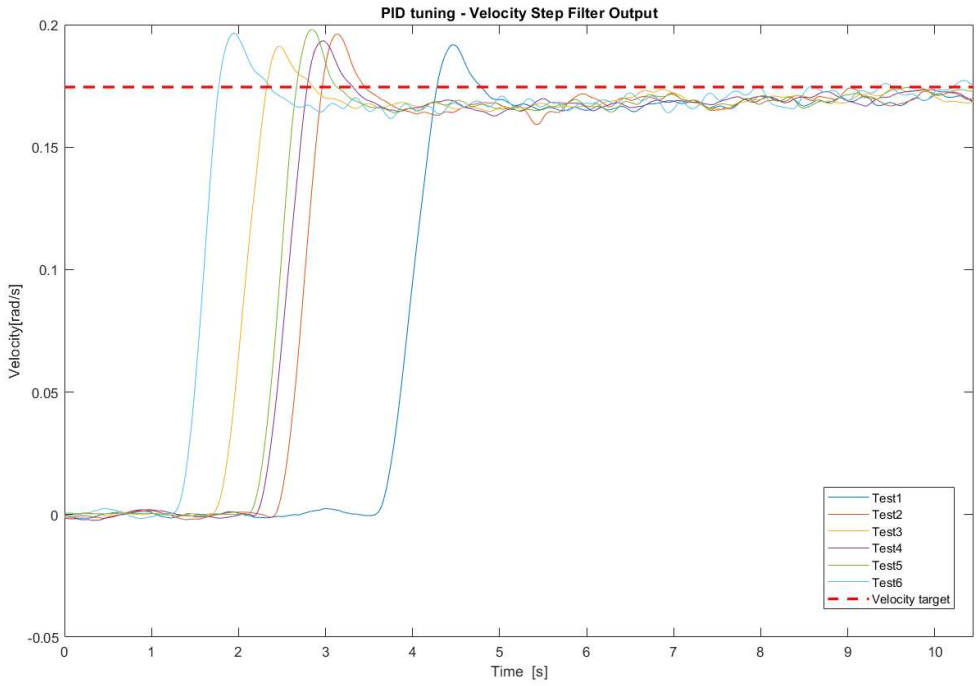
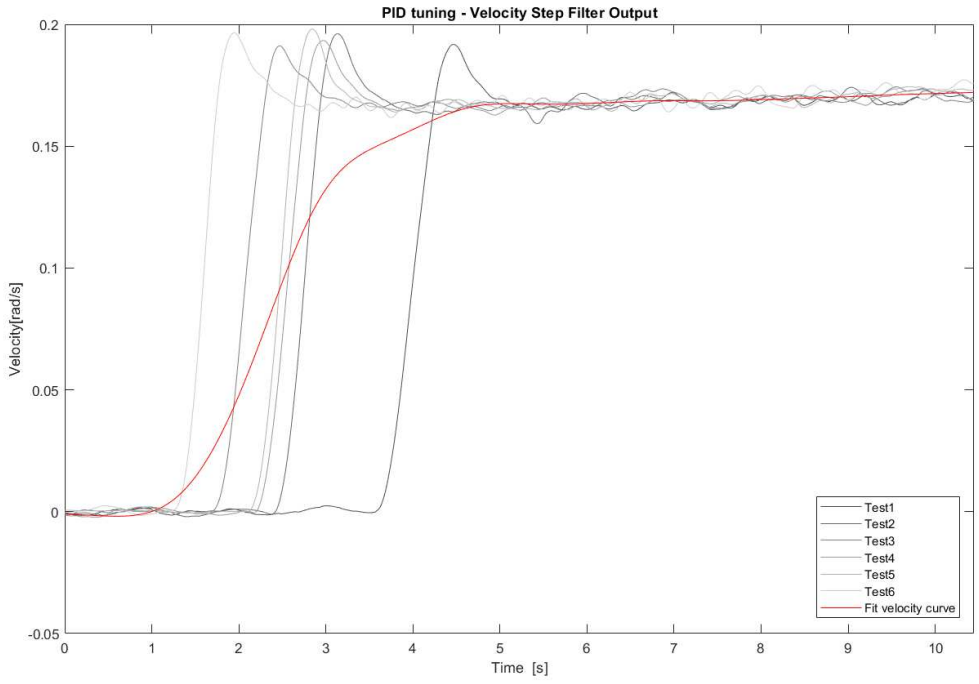


FIGURE 5.8: Velocity step output for velocity step input(The curves aren't aligned because the operator chooses the visualization prior).



(A) Color scale of velocity output with the target.



(B) Color scale of velocity output with the mean of signals.

FIGURE 5.9: Velocity step filtered output for velocity step input.

Looking at Figure 5.9(A) is possible to see the mean trend of the velocity step. The plateau value isn't the velocity target desired; this is the second error. This error consists of a delay between the target and the mean of the output, which is the effect of the Kd. The delay equals 0.006429 [rad/s] in angular degree is 0.3666929 [deg/s]. Considering the tendency to reduce this value in time we can appreciate the result and look for the position response. When we operate on the velocity the must condition is that the position increases without flexion point [Figure 5.10].

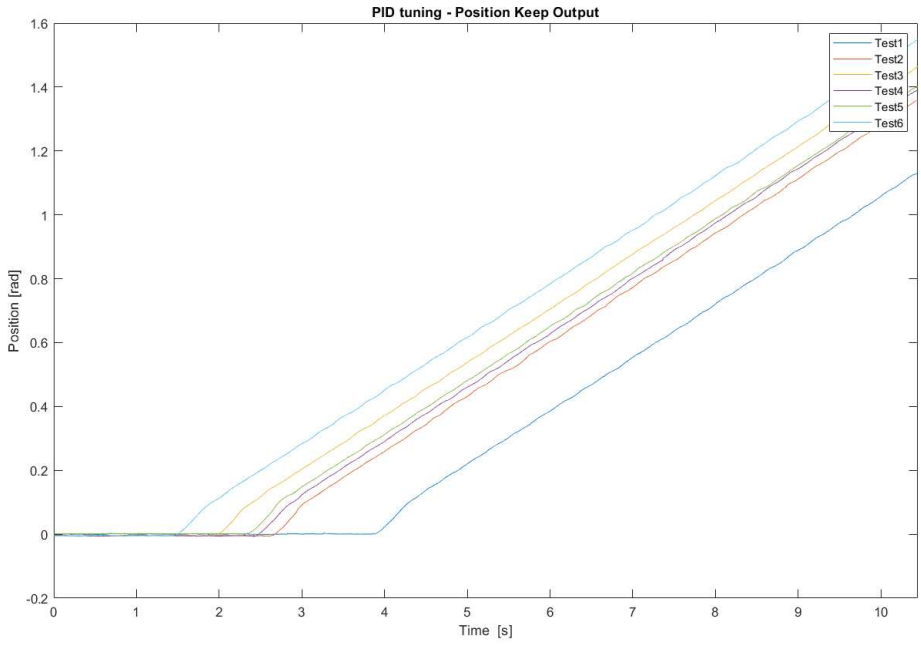
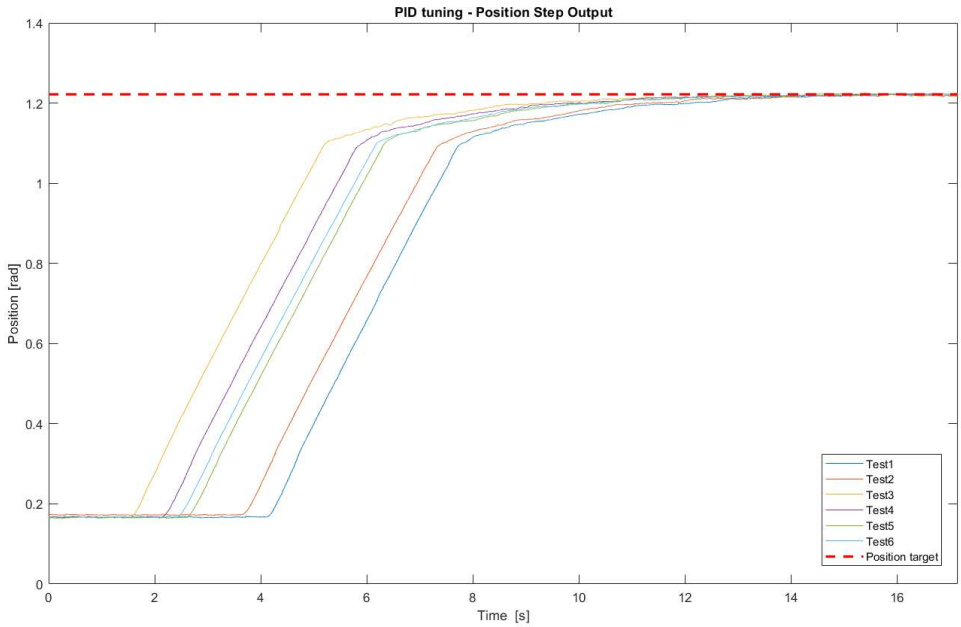


FIGURE 5.10: Slope and angular position of the joint during step velocity control.

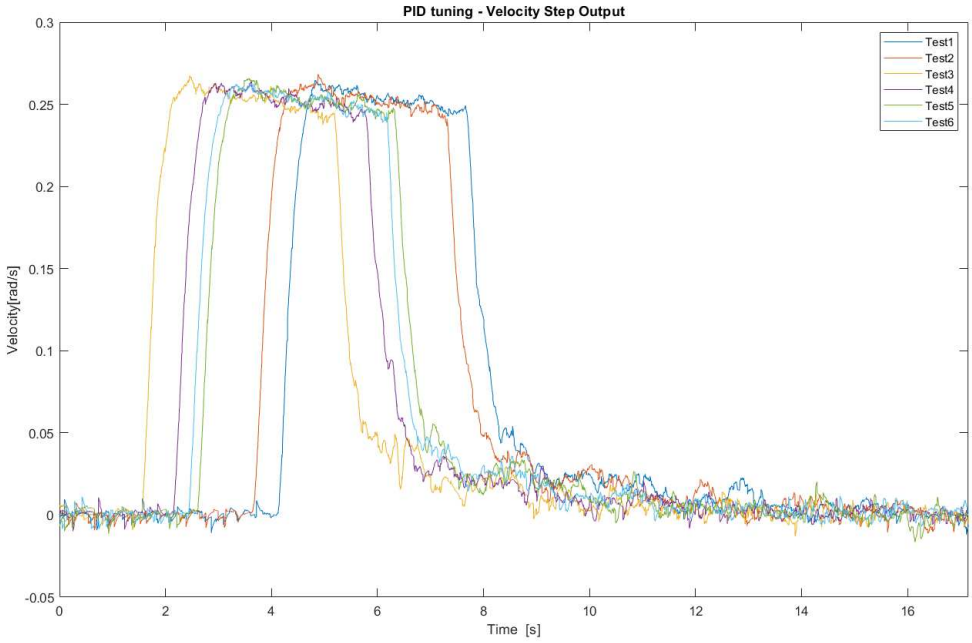
In this case, although the response of the speed is not a perfect step, the movement of the joint is smooth and without interruptions highlighting an intrinsic functioning.

Position control ring included

The same tests can be done in position control, activating all the control rings. The test consists of position step input from 0° to 10°,20°,30°... until we finish the angle range (the test is done also in the negative range of motion of the relative reference). I chose to represent the most significant step for the hip joint, the 70° flexion. This particular position is the most common during walking to overcome an obstacle or to rise steps. Moreover, in this specific angle of flexion, all the inertia is dumped into the hip joint.



(A) Step response for position step input with the target.



(B) Velocity output profile with position step input.

FIGURE 5.11: Output for a position step input.

So with a 70° flexion, we can evaluate the presence of labilities or some instabilities due to the mechanical system or to the control parameters. As can be seen in Figure 5.11(a), the step response is good in terms of readiness, and sensibility, the journey time between the the starting point and the plateau point is about 8 [sec]. The joint has done about 60° excursion in less than 10 seconds. Furthermore, the accuracy is higher than the velocity tests, joint "stops" very close to the target, with a delay of 0.00019 [rad], less than 0.01 [deg]. Unfortunately, also in this case is possible to see little oscillations around the target due to a velocity signal about zero but never zero [Figure 5.11(B)].

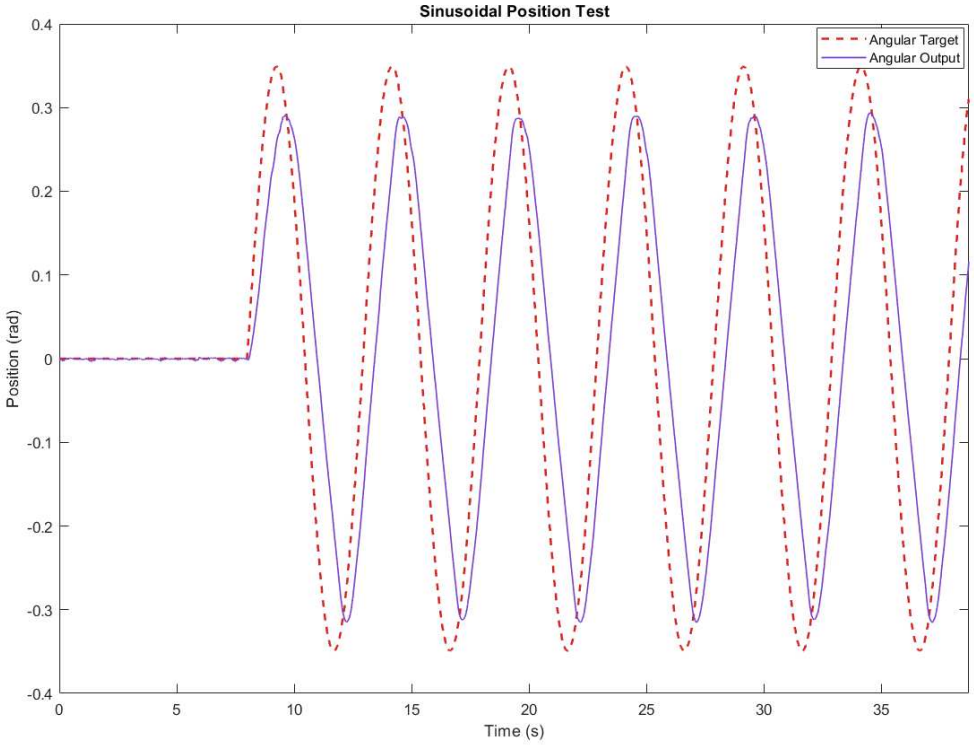
This phenomenon depends on the integrative effect of PID control, particularly for the Ki_v velocity dependant. The presence of noise in the background induces the storing of the history of the error, when this value is big enough to overcome the inertia of the system, the joint starts to move with a micro click. That error is partially a casual error due to interferences and noise of various types, commonly present in laboratories, and partially due to systematical error. The term systematical refers to a lower level of filtering and the absence of a powerful microprocessor, that can do sample analysis at a high rate.

5.3 Dynamic control test

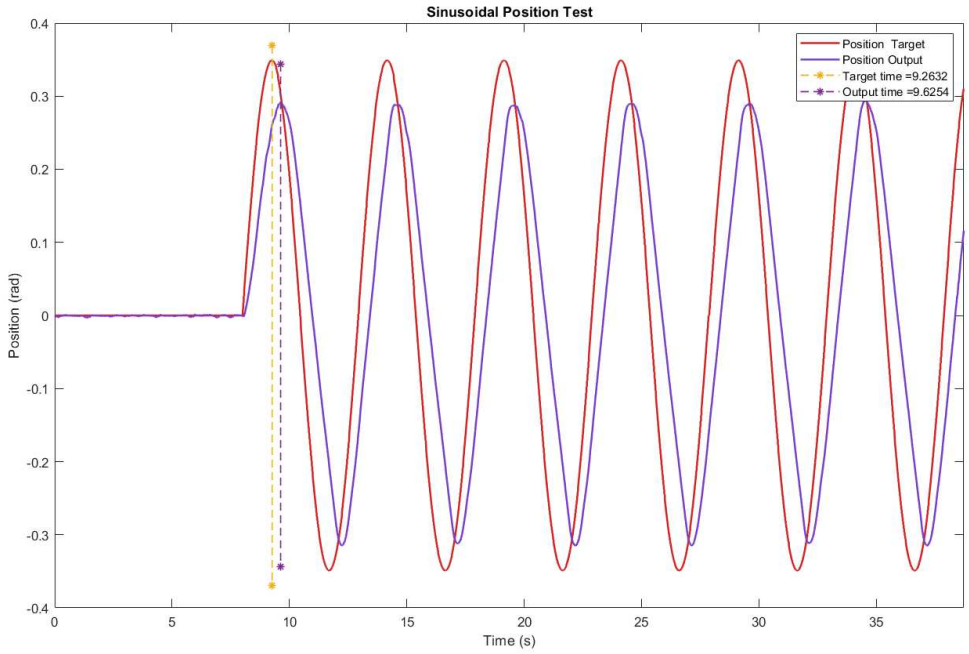
As outlined in the previous chapter, the main part of the project is the dynamic control during the walking gait. The tests were carried out with the same experimental setup paying attention to the movement excursion rate. Our attention was focused on the ability of the system to follow a complex command as a sinusoidal input without lag, incrementation, or decline of the position signal, and the absence of spike movement. The scattering represents the challenging part of the tests during the cycle or the presence of systematic errors due to the PID control and setup. The results with the knee are presented in the AppendixD.

5.3.1 Sinusoidal input

As has been done for the step input, we introduced a complex sinusoid curve as input. The mid-level control elaborates the curves and sends to the lower-level control a succession of points to reproduce the native curve. When the joint receives the first point target, it starts to move and follows all the directions sent by the control system. It's important to understand that the system will be always late rather than the control input. That's because the readiness of the system is never 100%. Also with strong control and accurate tuning of the parameters, it's impossible to completely synchronize the two curves. Also in this case is necessary to improve a tuning procedure to achieve the better work of the system.



(A) Angular output with the subsequent angular target.



(B) Evidence of the time lag between the two curves.

FIGURE 5.12: Sinusoidal output for position control test first iteration.

The precedent figures show the position output with a sinusoidal input control. It's evident how the joint movement is late compared to the input signal, the lag can be calculated with a simple subtraction between the time of the two apex. The lag value is about 0.3622 seconds, 1/3 of a second [Figure 5.12(B)]. There is also another invalidating condition, the few accuracy of the output signal. The maximum value of the target apex is 0.3491 rad, which means 20 degrees. The output maximum value is about 0.2490 rad, which means 14.3 degrees; there is a difference of about 6 degrees between the curves. The reasons for these differences can be explained by going deep into the analysis and looking at the velocity profile [Figure 5.13].

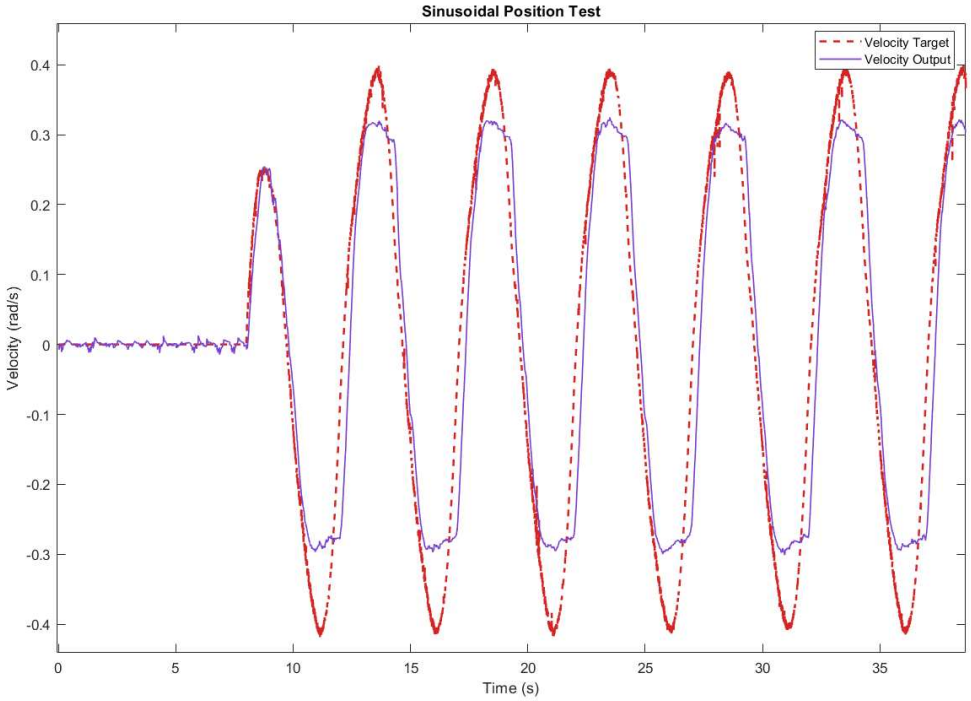
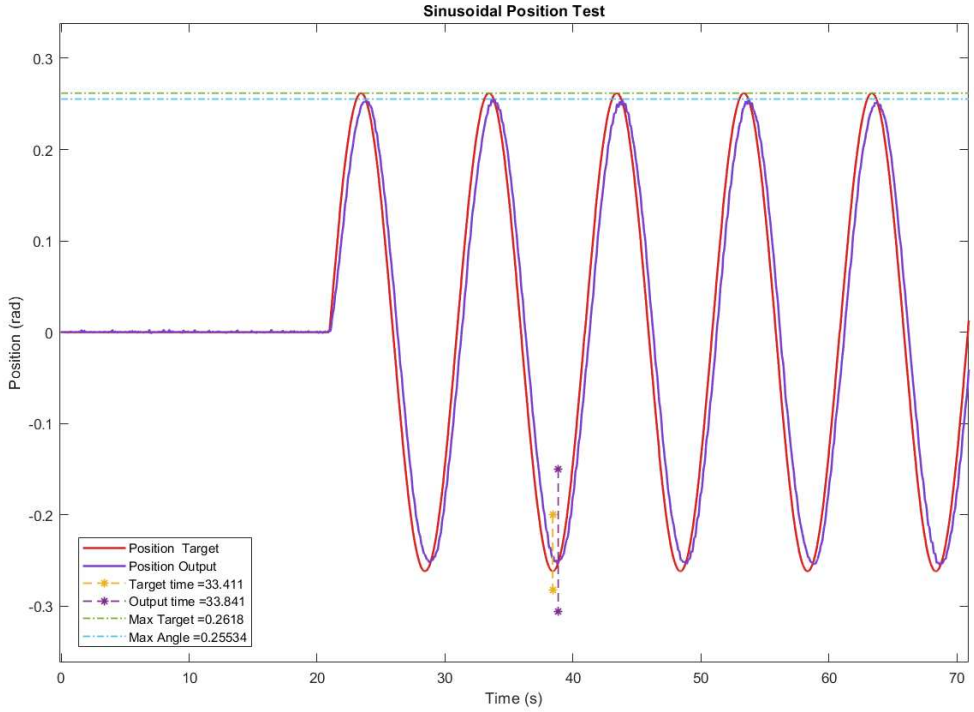
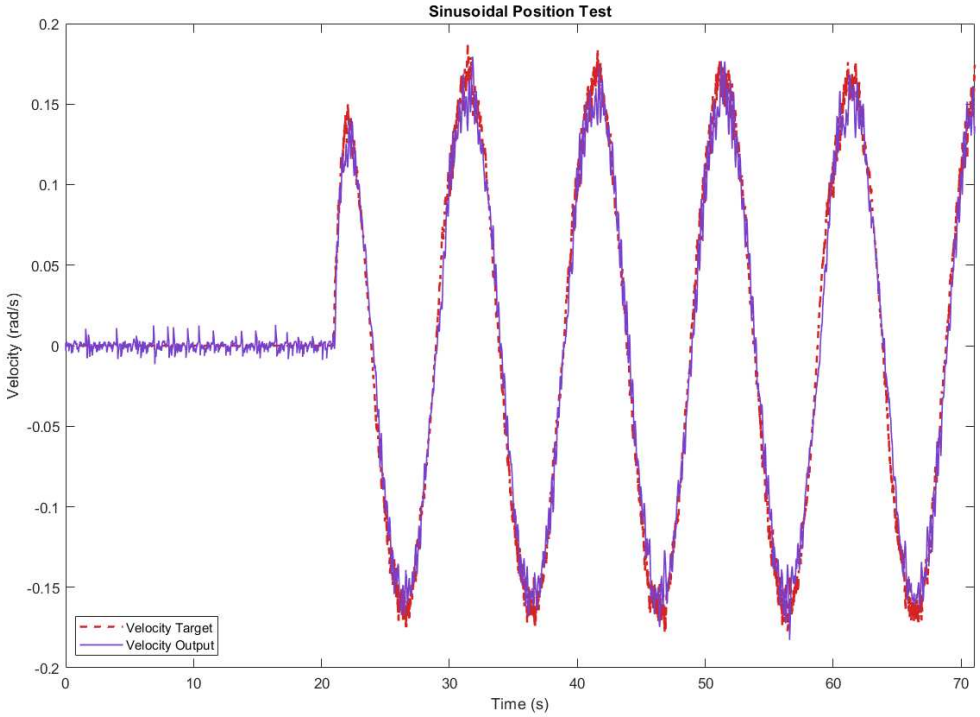


FIGURE 5.13: Velocity output with velocity input for a position sinusoidal control.

It's evidence that the velocity can't perform completely the profile, the input signal is too fast and the joint's execution is too conservative to operate a run. We had to increase the command system parameters and force the velocity to follow the correct profile as better can be done with this type of system [Figure 5.14]. After that, the results show a higher accuracy, the difference between the apex is about 0.01 rad, 0.57 degrees; but there are no significant differences in the time lag, which remains 0.35 seconds [Figure 5.14].



(A) Angular output with the subsequent angular target and evidence of the time lag between the two curves.



(B) Velocity output with velocity input for a position sinusoidal control.

FIGURE 5.14: Sinusoidal output for position control test last iteration.

5.3.2 Gait profile input

In this section, we will go deep into the analysis of the system with a complex input born from the knowledge of physiological curves profile. I will show the position output of the hip joint and also the output of the knee joint. The parameters used before were maintained also in these tests. Firstly I want to show and comment on the system response in terms of precision and repeatability.

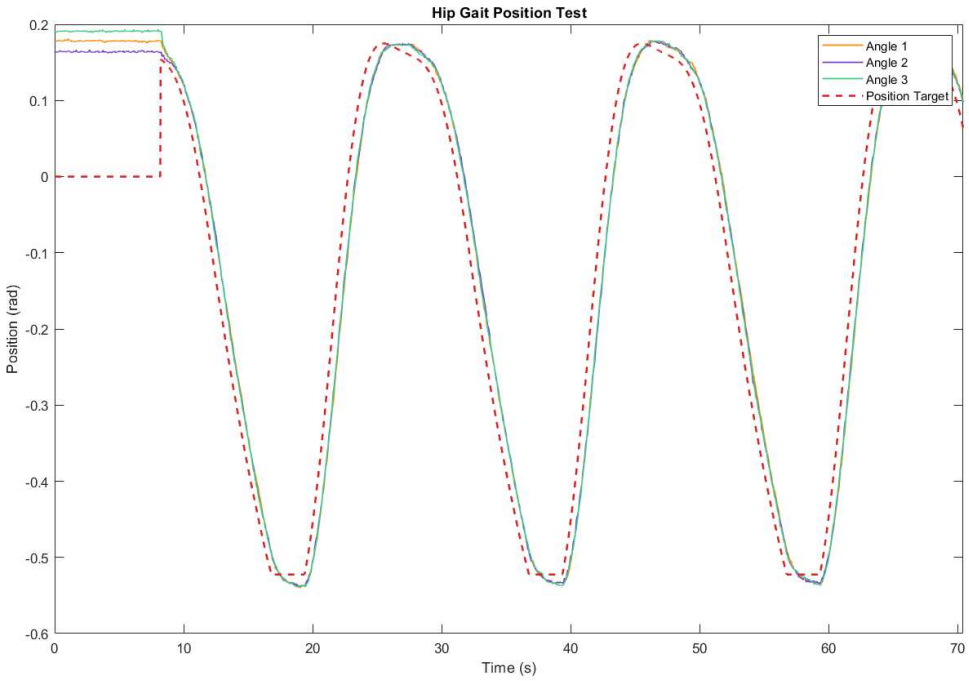
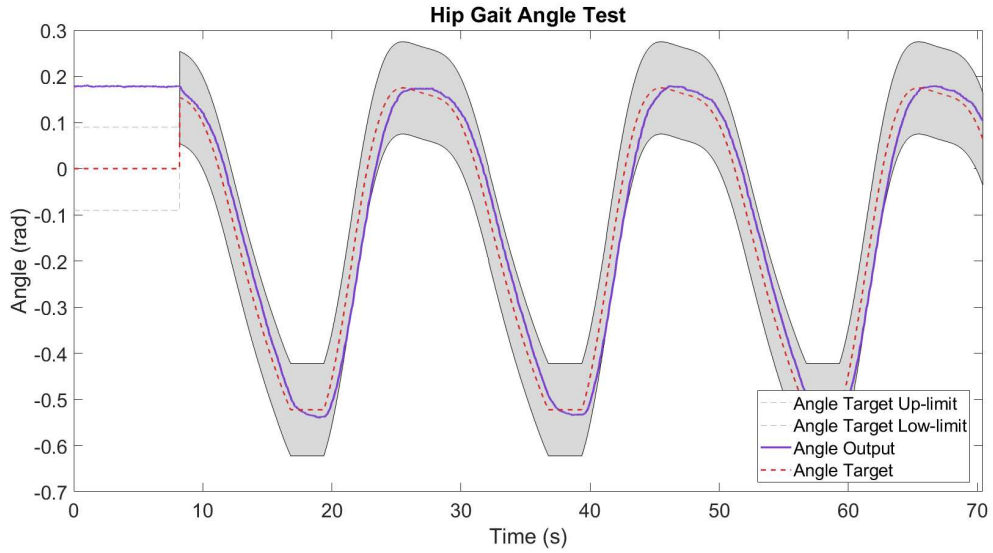


FIGURE 5.15: Angular gait profile for the hip joint and input target.

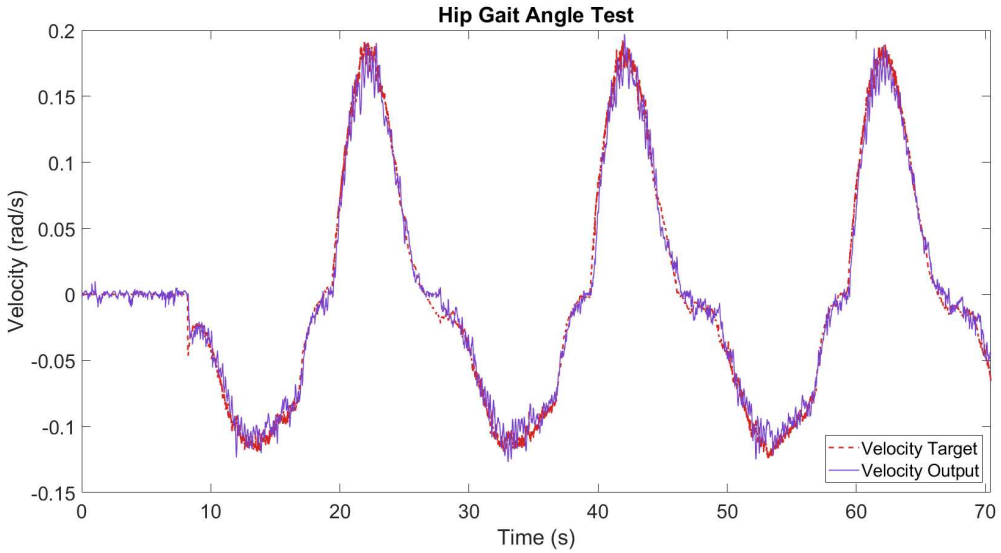
As can be seen in Figure 5.15, the joint response is always the same also when the starting point is different, this evidences an implicit good regulation and a solid functioning. This is evidence of high precision. Meanwhile, if we look at the accuracy of the system we can see the presence of a little overshot on the bottom of the curves. This effect can be made by the combination of two things:

- The presence of the force of inertia in agreement with the direction of movement, that increases the velocity;
- A slowly response due to the lag between the signal and the command.

If we want to increase or decrease one of these two effects, we will have an opposite change in the other one. With it, we can only mediate between the two aspects.

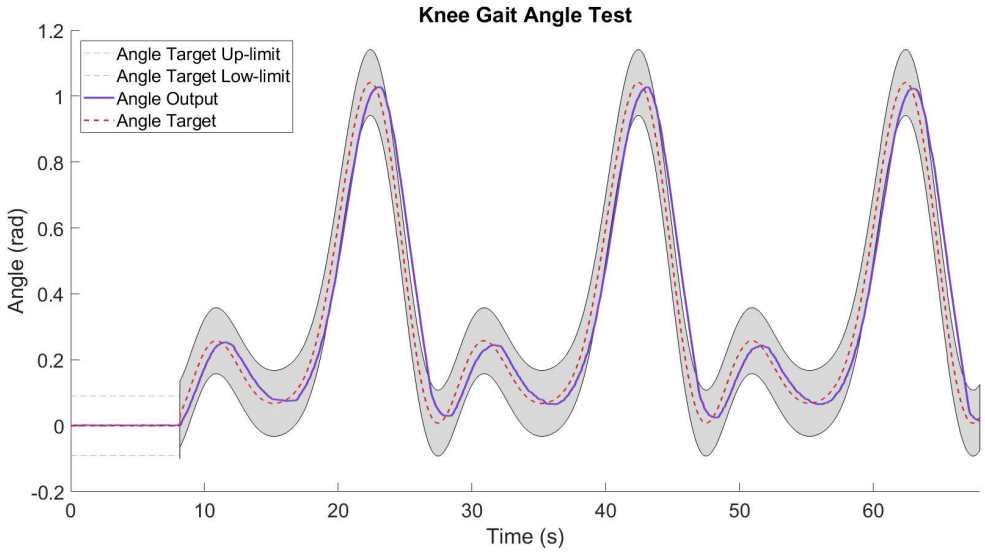


(A) Angular gait profile for the hip joint and physiological band.

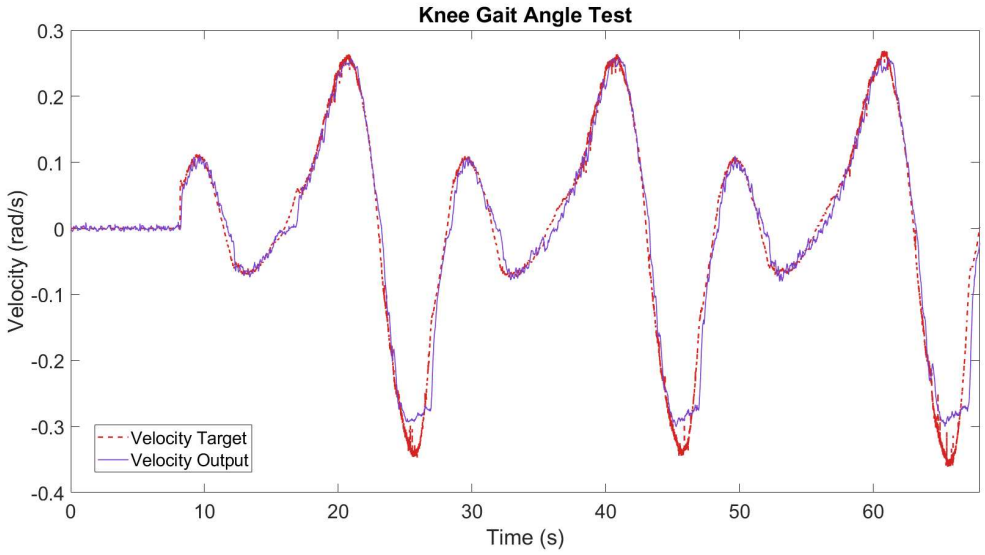


(B) Velocity gait profile for the hip joint and input target.

FIGURE 5.16: Gait profile for the hip joint and input target.



(A) Angular gait profile for the knee joint and physiological band.



(B) Velocity gait profile for the knee joint and input target.

FIGURE 5.17: Gait profile for the knee joint and input target.

Moving now on to another aspect, the mean joint movement can be observed from the above curves with the lower and upper extremes of the physiological spectrum of movement for the hip and knees indicated. It is possible to observe how the average curve, in almost all moments, falls within the desired spectrum [Figure 5.16(a)][Figure 5.17(a)]. This is a reassuring result as it proves the functionality in terms of the physiology of the device. Nevertheless, there are still small moments in which the curve comes out of the spectrum. Observing the speed we can understand why [Figure 5.16(b)][Figure 5.17(b)]. The signal in speed does not follow perfectly the target speed. This operation can be pushed further by improving the overall result but at the expense of possible and subsequent dyssynchrony of the system.

The knee seems to be particularly affected by this problem, which evidences the necessity to modify the parameters of control to ensure higher stability during the target follow. In this case, the only possibility to ensure a correct output is to differentiate the two targets of control by a switch of the parameters.

5.4 Antrophometric comfort test

This section is dedicated to the comfort feel of the users. It is fundamental for that type of system to ensure always the comfort of the patient that will use the EXO. Was largely showed how a bad design, uncomfortable features, and complex interfaces lead to the failure of the system. The users don't want to walk with an uncomfortable system every day for a long period, and as all the other people do, they are interested in their aspect. Comfort is a keyword during the mechanical design and choice. The aspects explored in this project are:

- The comfort during the rest phase;
- The comfort during the step phase;

The tests were carried out by a volunteer wearing the exoskeleton during the various phases. All the wiring systems were disconnected and elements born to equip the system on the users insured on the legs of the volunteer. The sliding covers were regulated on the correct side of things and legs to ensure the joints, we used strips to bind the legs of the users and pillows to fill the cover [Figure 5.18(A)][Figure 5.18(B)].

(A) Front view of the EXO (B) Users with the EXO wore.

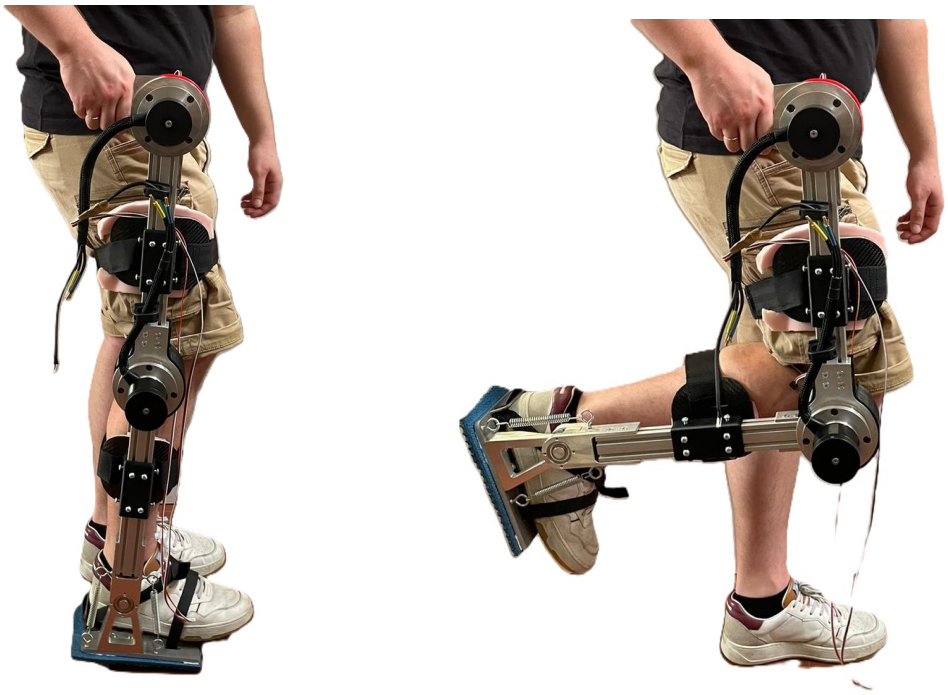
FIGURE 5.18: EXO system and volunteer.

The report on device comfort can be summarised by three main aspects:

- The device's clothing
- The dynamics of movement
- The fatigue during use

Emulating the case of a person who cannot move his lower limbs, if not to a limited extent, the volunteer simulated the process by which he attaches the device to his legs. During the course, he highlighted how the joining systems are resistant and comfortable but not very tight because concentrated in a narrow area. This was noticed both during the very first movement of wearable stand-up and during movements related to step. It was, however, highlighted that the process was intuitive and fast. This is thanks to the velcro straps that make the process itself quick and lower the workload [Figure 5.19(A)][Figure 5.19(B)].

During the various attempts of movement by the volunteer it turned out that only through the correct synchronization of movements and a correct arrangement of the artificial joints in correspondence with the natural joints was possible to perform complete actions without difficulty. The EXO was tested with the motors switched off. In this case, the system is in transparent mode, that is engine turned off but with back-drivable joints. In some cases the



(A) Lateral view of the EXO's leg.

(B) Lateral view of the EXO's leg during flexion.

FIGURE 5.19: EXO system and volunteer during movements.

system could move on the user's legs and that results in a mismatch between the joints, this can cause fragmented movement due to the necessity of adjusting manually the joints. This is partly due to the joint's lability, which will not be completely bound to the leg. It's a small price to pay to ensure maximum usability by users with different anthropometry.

Regarding the fatigue associated with the use of the device, the user expressed a positive comment in favor of the reduced footprint and weight of the device by communicating how repeating movements while wearing the exoskeleton did not result in high motor load. Rather, the effort was due to the user in the time-of-time synchronizing the movement to make it as smooth as possible. The work stress to which the volunteer was thus subjected was mainly related to the process of adaptation of the volunteer to movements with the device.

Chapter 6

Conclusion

The main objective of this work was to assemble, check, and validate the prototype of the lower limb exoskeleton at the DII Machine Building Laboratory in collaboration with the Autonomous Intelligent Laboratory (IAS-Lab) group of the University of Padova. Along the way, I have faced various problems which have led to new solutions being studied. This concluding chapter will then comment on the steps taken during this experience. Finally, some possible future developments of the project will be outlined.

6.1 Discussion and evidences

Material and design
The material and geometries used for the construction of the device, in particular the frame part, have given excellent results in terms of strength, adaptability, and safety. The use of aluminum structures was advantageous both from an economic point of view and from the point of view of reducing the overall weight of the device. A lighter device was considered more comfortable and above all less cumbersome during movements and maneuvers. Although there are few examples in the literature of devices with a structure made entirely of aluminum, this is proof that the use of advanced alloys born natively in the industrial field can be a valuable asset also in the medical field.

The design developed through adaptable profiles allows the use of the for different users with very different sizes and dimensions, all thanks to a reasonable modification of the structure easily achievable by an external operator. The slim design allows the user to move easily in even the harshest environments and wear the device in almost any context. The strengths of the design and materials implemented for the prototype development of the lower limb skeleton will be:

Advantages

Low production costs

Excellent balance between strength and weight

Presence of spaces for the equipment of other devices

Easy adaptation to different anthropometric measurements

Sensor and attuation system

The actuation and transmission systems implemented were particularly effective for the type of control carried out. The high-capacity motor can guarantee an excellent speed of execution of the movements and precise control. The gearbox in turn allows a smooth and direct movement from the motor to the joint and the possibility of reversing the motion without delay. The implementation type ensures the use of battery systems for autonomous navigation or fixed power supply for testing and future implementations. Consumption was lower than that of many other similar devices, further confirming the cost-effectiveness of the device.

Advantages

Excellent balance between dimension and torque

Easy adaptation and speed reversing of motion

Connection compatibility

Low costs

The system for acquiring feedback on position and speed was found to be perfectly compatible with the mechanical device and accurate in the measurements. The reading speed, accuracy, and simplicity of programming allowed the encoder to be easily implemented. The magnetic sensor has demonstrated a solid resistance to environmental interference and system vibration, always ensuring reliable reading. The small dimensions and easy assembly make it an effective alternative to the market's rival encoders.

Advantages

High accuracy

Intrinsically stable

Fast read of position

Small size of accommodation

Readiness and accuracy

All the tests done on the lower-level control demonstrated that the system can achieve simple commands like a single step or maintain a position target. It was largely shown that the low system control can follow point references with great readiness and more than sufficient accuracy. The maximum speed achieved by the system is congruent to the physiological

velocity imposed by the normal step gait, suggesting that the system can perform a correct walk gait also with a fast pace. The repetitiveness of the results and the solid tendency of the system show that the low-level control with a SimpleFOC library is optimal for that type of application.

The analysis of the mid-level system shows that the system is particularly effective in performing complex profiles of position and point speed, thus highlighting the exoskeleton's ability to emulate the native walking gate. These results were obtained for a single leg although the operation is symmetrical and therefore we are responsible for the same also for the remaining.

Advantages

High precision and accuracy

Physiological position profile

Fast velocity

Low level of noise

Comfort and wearing

The device in its entirety has shown excellent comfort and above all an excellent adaptability to the measurements and dimensions of the user. The user praised the ability to adapt extruded profiles to their own limb height and length. In addition, the simplicity and weight of the device were a key point that makes the exoskeleton easier to wear and bear once worn.

6.2 Limits

Material and design

The limits resulting from the choice of basic material such as aluminum, the presence of adjustable joints, and a low-weight device are due to the resistance that this last one has but above all to the bending phenomena to which the elements will be subjected. It is easy to demonstrate how the implemented structures will have a certain degree of compliance related to the material and type of connection.

This inevitably affects the user's comfort, but especially the control and execution of movements. Although as mentioned above oscillation of the pelvis reflecting the swinging movement of the native hip is a positive phenomenon; the bending in the presence of joints implemented along only the sagittal plane can modify the trajectory desired. During various tests for the position control, difficulties arose in obtaining a signal without noise, part of

which noise was also caused by the oscillation of the experimental setup is of the device.

Sensor and attuation system

The negative aspect of using a single shaft system, direct actuation, and in the absence of an engine brake is to have to continuously supply current to generate, even during moments of rest, a minimum support torque. This support pair must be able to both ensure static in the upright position and ensure elevation and maintenance of the limb during complex or prolonged movements. It was experimentally possible to see how the combined system of motor and control has determined a residual vibration linked to the alternation of opposite-direction couples that cancel each other and generate an ideally zero movement.

The magnetic sensor during the various tests didn't show any type of disadvantages, except one, the noise due to the connection cable system. The sensor chip was born with an out-plug connection for a cable bridge of a small length. We had to adapt the sensor to our aim with cable extension and a different plug system. Unfortunately, these types of changes modify the entity of the signal, including background noise.

Readiness and accuracy

The results revealed several negative aspects regarding the control system and the execution of commands. It is evident that the system, although having good accuracy and precision, is subject to unstable conditions. Background noise, electromagnetic signals, or simply high mechanical tolerances can cause incorrect joint positioning; both for point controls and complex controls such as profiles. In addition, what made the thesis process more difficult was the fact that a feedback system had a different reading and execution speed from the central control computer. This has resulted in many cases in the presence of phenomena called Spike, that is, moments when there is a de-synchronization between the command output and the reading input.

The only problem with the profile execution is that the system updates at a low rate. That is, if the setting of parameters is such as to push the control and therefore have a very accurate and sudden response can happen that during an update cycle of the command you go beyond the command itself inducing a slowdown and then imposing a time lag Between command and execution.

6.3 Future developments

The first aspect to be addressed is certainly the updating of the system hardware systems, in particular the central computer and the drivers running the engines. This would make it possible to compensate for all the shortcomings arising from a different update and read rate between the central computer and the magnetic sensor.

Complete the elements of protection and support of the user. Develop then, always using PLA, all remaining covers for the rotating parts and cover for the points of contact with the user to ensure maximum comfort is maximum safety.

Perform multiple joints simultaneously employing a system based on CanBUS. This would make it possible to study the total current consumption, the control system's ability to synchronize the various joints, and also to analyze stability during complex runs. This is a fundamental step in understanding what kind of portable power system the machine needs.

As far as the analyses carried out are concerned, it would be essential to evaluate the effect of the system during a gait cycle utilizing motion capture. This is an essential examination to determine if the execution of the joints is really what we read through the sensor and if the execution results are once again physiological. Motion capture readings can also improve the setting of parameters, as it would be possible to see actual deviations from the range of physiology or unwanted phenomena.

Develop tests for load using weights or a dummy. The system as it is designed should be able to react in a way almost proportional to the weight agent. But since operation is always related to a range of allowable weights, you may need a dedicated setting for much higher ranges in terms of weight.

Appendix A

Tuning of angular range for right Hip

```
1
2  //    FILE: AS5600_outmode_analog_100.ino
3  // PURPOSE: experimental demo
4
5  //Connect the OUT pin to the analog port of the processor
6  #include <ros.h>
7  #include <std_msgs/Float32MultiArray.h>
8  //#include <std_msgs/Float32.h>
9  #include "lowpass_filter.h"
10 #include "AS5600.h"
11 #include "Wire.h"
12
13 //ROS inizialization
14 ros::NodeHandle  nh;
15
16 AS5600 as5600;   //  use default Wire
17
18 //////// ROS Publishers ///////////
19 std_msgs::Float32MultiArray angle_msg;
20
21 ros::Publisher pub_angle("/angle", &angle_msg);
22
23
24 // Sensor calibration
25 float RH_minRaw = 132.1875;
26 float RH_maxRaw = -0.3515625;
27 float RH_minAng = -37.54;
28 float RH_maxAng = 95.0;
29
30 //Value to controll time lop
31 //float tic = 0.0;
32 //float elapsed = 0.0;
33
```

```
34
35 #define ANGLE_TIME_CONSTANT 0.8
36 #define VEL_TIME_CONSTANT 1.0
37 LowPassFilter AngleLowPassFilter(ANGLE_TIME_CONSTANT);
38 LowPassFilter VelocityLowPassFilter(VEL_TIME_CONSTANT);
39
40
41 float previousAngle = 0.0;
42 float velocity = 0.0;
43 float min_elapsed_time = 0.00100;
44 unsigned long previousTime = 0;
45
46
47 void setup()
48 {
49
50   Wire.begin();
51
52   as5600.begin();  //  set direction pin.
53   as5600.setDirection(0);  // default, just be explicit.
54   as5600.setOutputMode(AS5600_OUTMODE_ANALOG_100);
55
56   //for rosserial
57   nh.getHardware()->setBaud(115200);
58   nh.initNode();
59   nh.advertise(pub_angle);
60
61   while(!nh.connected()) nh.spinOnce();
62   delay(1000);
63 }
64
65 void loop()
66 {
67   // Lettura del tempo
68   //tic = micros();
69
70   // Read the current analog value
71   int analogValueF = analogRead(A1);
72
73   // Convert the filtered analog value to degrees
74   float degreeF = analogValueF * 0.3515625;
75
76   if (degreeF > 180.0)
77   {
```

```
78      degreeF = degreeF - 360;
79    }
80
81    // Apply the low-pass filter
82    float filteredValue = AngleLowPassFilter(degreeF);
83
84    // Applica l'offset durante la mappatura
85    float mappedDegreeF = mapToFraction(filteredValue, RH_minRaw,
       RH_maxRaw , RH_minAng, RH_maxAng);
86
87    // Ensure that mappedDegree remains in the range 0-120
88    //mappedDegreeF = fmod(mappedDegreeF,);
89
90    // Convert the final angle to radians
91    //float radians = mappedDegreeF * (PI / 180.0);
92    //float radians = mappedDegreeF;
93
94    // Calculate the time difference
95    unsigned long currentTime = micros();
96    float delta_angle = mappedDegreeF - previousAngle;
97    float deltaTime = (currentTime - previousTime) * 1e-6f; // Convert
       microseconds to seconds
98
99    // Calculate the angular velocity
100   if(deltaTime < min_elapsed_time) return velocity;
101   else velocity = delta_angle / deltaTime;
102
103 velocity = VelocityLowPassFilter(velocity);
104
105   // Update previous values for the next iteration
106   previousAngle = mappedDegreeF;
107   previousTime = currentTime;
108
109
110
111   float pub_data[2] = {mappedDegreeF, velocity};
112   angle_msg.data = pub_data;
113   angle_msg.data_length=2;
114   pub_angle.publish(&angle_msg);
115
116
117   nh.spinOnce();
118
119   delay(1);
```

```
120   //elapsed = micros() - tic;
121 }
122
123 float mapToFraction(float x, float in_min, float in_max, float out_min
      , float out_max) {
124   // Map x from in_min to in_max to a fraction between out_min and
      out_max
125   return (x - in_min) * (out_max - out_min) / (in_max - in_min) +
      out_min;
126 }
```

Appendix B

Tuning of angular range for right Knee

```
//    FILE: RIGHT_KNEE.ino
// PURPOSE: experimental demo

//  connect the OUT pin to the analog port of the processor
#include <ros.h>
#include <std_msgs/Float32MultiArray.h>
//#include <std_msgs/Float32.h>
#include "lowpass_filter.h"
#include "lowpass_filter_velocity.h"
#include "AS5600.h"
#include "Wire.h"

//ROS inizialization
ros::NodeHandle  nh;

AS5600 as5600;   //  use default Wire

//////// ROS Publishers ///////////
std_msgs::Float32MultiArray angle_msg;

ros::Publisher pub_angle("/angle", &angle_msg);


// Sensor calibration
float LK_minRaw = 265.0;
float LK_maxRaw = 357.0;
float LK_minAng = 0.0;
float LK_maxAng = 93.0;

//Value to controll time lop
//float tic = 0.0;
//float elapsed = 0.0;

```

```
162
163 #define TIME_CONSTANT 0.1
164 LowPassFilter lowPassFilter(TIME_CONSTANT);
165 LowPassFilter lowPassFilter_velocity(TIME_CONSTANT);
166
167 float previousAngle = 0.0;
168 float velocity = 0.0;
169 float min_elapsed_time = 0.000100;
170 unsigned long previousTime = 0;
171
172
173 void setup()
174 {
175   //Serial.begin(115200);
176   //Serial.println(__FILE__);
177   //Serial.print("AS5600_LIB_VERSION: ");
178   //Serial.println(AS5600_LIB_VERSION);
179
180   Wire.begin();
181
182   as5600.begin();  //  set direction pin.
183   as5600.setDirection(0);  // default, just be explicit.
184   as5600.setOutputMode(AS5600_OUTMODE_ANALOG_100);
185
186   //for rosserial
187   nh.getHardware()->setBaud(115200);
188   nh.initNode();
189   nh.advertise(pub_angle);
190
191   while(!nh.connected()) nh.spinOnce();
192   delay(1000);
193 }
194
195 void loop()
196 {
197   // Lettura del tempo
198   //tic = micros();
199
200   // Read the current analog value
201   int analogValueF = analogRead(A0);
202
203   // Convert the filtered analog value to degrees
204   float degreeF = analogValueF * 0.3515625;
205
```

```
206   // Apply the low-pass filter
207   float filteredValue = lowPassFilter(degreeF);
208
209   // Applica l'offset durante la mappatura
210   float mappedDegreeF = mapToFraction(filteredValue, LK_minRaw,
       LK_maxRaw , LK_minAng, LK_maxAng);
211
212   // Ensure that mappedDegree remains in the range 0-120
213   mappedDegreeF = fmod(mappedDegreeF, 120.0);
214
215   // Convert the final angle to radians
216   float radians = mappedDegreeF * (PI / 180.0);
217
218   // Calculate the time difference
219   unsigned long currentTime = micros();
220   float delta_angle = radians - previousAngle;
221   float deltaTime = (currentTime - previousTime) * 1e-6f; // Convert
       microseconds to seconds
222
223   // Calculate the angular velocity
224   if(deltaTime < min_elapsed_time) return velocity;
225   else velocity = delta_angle / deltaTime;
226
227   float velocity = lowPassFilter_velocity(velocity);
228
229   // Update previous values for the next iteration
230   previousAngle = radians;
231   previousTime = currentTime;
232
233
234
235   float pub_data[2] = {radians, velocity};
236   angle_msg.data = pub_data;
237   angle_msg.data_length=2;
238   pub_angle.publish(&angle_msg);
239
240
241   nh.spinOnce();
242
243   delay(1);
244   //elapsed = micros() - tic;
245 }
246
```

```
247 float mapToFraction(float x, float in_min, float in_max, float out_min
        , float out_max) {
248   // Map x from in_min to in_max to a fraction between out_min and
        out_max
249   return (x - in_min) * (out_max - out_min) / (in_max - in_min) +
        out_min;
250 }
```

Appendix C

Gait trajectory generation

```
253
254 #!/usr/bin/env python
255
256 import numpy as np
257 import matplotlib.pyplot as plt
258
259 class trajectory_generator:
260   def __init__(self, cycle_duration, fs, gear_ratio, driver, h_offset
     = 29.1, k_offset = 24.127):
261     self.cycle_duration = cycle_duration
262     self.fs = fs
263     self.w = np.array([0, 2, 4, 6])/cycle_duration
264     self.c = [0.208, 0.362, -0.066, 0.001, 0.766, -0.099, -0.219,
    0.008]
265     self.f = [np.nan, -0.103, -0.010, 0.029, -0.342, 0.168, 0.084]
266     self.h_offset = np.deg2rad(h_offset)  # [rad]
267     self.k_offset = np.deg2rad(k_offset) # [rad]  # 26.127
268     self.cycle_time = np.linspace(0, cycle_duration, cycle_duration*fs
    )
269     self.gear_ratio = gear_ratio
270     self.driver = driver
271     self.t = 0
272     self.prova = []
273
274     self.A = -0.2
275     self.torque_freq = 0.5
276     self.step_perc = 0.5
277     self.read_array = []
278     self.write_array = []
279
280     self.pre = 0.0
281
282   def get_hip_angle(self,t):
```

```
283     qh = self.c[0]*np.cos(self.w[0]*np.pi*t) + self.c[1]*np.cos(self.w
        [1]*np.pi*t) + self.f[1]*np.sin(self.w[1]*np.pi*t) + self.c[2]*np.
        cos(self.w[2]*np.pi*t) + self.f[2]*np.sin(self.w[2]*np.pi*t) +
        self.c[3]*np.cos(self.w[3]*np.pi*t) + self.f[3]*np.sin(self.w[3]*
        np.pi*t) - self.h_offset    # [rad]
284     return qh
285
286   def get_knee_angle(self,t):
287     qk = self.c[4]*np.cos(self.w[0]*np.pi*t) + self.c[5]*np.cos(self.w
        [1]*np.pi*t) + self.f[4]*np.sin(self.w[1]*np.pi*t) + self.c[6]*np.
        cos(self.w[2]*np.pi*t) + self.f[5]*np.sin(self.w[2]*np.pi*t) +
        self.c[7]*np.cos(self.w[3]*np.pi*t) + self.f[6]*np.sin(self.w[3]*
        np.pi*t) - self.k_offset    # [rad]
288     return qk
289
290   def get_hip_velocity(self,t):
291     qdoth = - self.c[0]*np.sin(self.w[0]*np.pi*t) - self.c[1]*np.sin(
        self.w[1]*np.pi*t) + self.f[1]*np.cos(self.w[1]*np.pi*t) - self.c
        [2]*np.sin(self.w[2]*np.pi*t) + self.f[2]*np.cos(self.w[2]*np.pi*t
        ) - self.c[3]*np.sin(self.w[3]*np.pi*t) + self.f[3]*np.cos(self.w
        [3]*np.pi*t)    # [rad/s]
292     return qdoth
293
294   def get_knee_velocity(self,t):
295     qdotk = - self.c[4]*np.sin(self.w[0]*np.pi*t) - self.c[5]*np.sin(
        self.w[1]*np.pi*t) + self.f[4]*np.cos(self.w[1]*np.pi*t) - self.c
        [6]*np.sin(self.w[2]*np.pi*t) + self.f[5]*np.cos(self.w[2]*np.pi*t
        ) - self.c[7]*np.sin(self.w[3]*np.pi*t) + self.f[6]*np.cos(self.w
        [3]*np.pi*t)    # [rad/s]
296     return qdotk
297
298   ## Riferimento coppia sinusoidale
299   def get_torque_sin(self,t):
300     torque_val = self.A*np.sin(self.torque_freq*np.pi*2*t)
301
302     return torque_val
303
304   ## Riferimento  gradino
305   def get_torque_step(self,t,perc):
306     if t < perc*self.cycle_duration:
307       return 0.0
308     else:
309       return self.A
310
```

```
311  def get_cycle_time(self):
312    return self.cycle_time[self.t]
313
314  def increment_time(self):
315    self.t = int((self.t + 1) % (self.cycle_duration*self.fs))
316
317  def plot_knee_cycle(self):
318    fig, axs = plt.subplots(2)
319    axs[0].plot(self.cycle_time, np.rad2deg(self.get_hip_angle(self.
     cycle_time)))
320    axs[0].set(xlabel='time [s]', ylabel='angle [  ]', title='Hip
     trajectory')
321    axs[1].plot(self.cycle_time, np.rad2deg(self.get_hip_velocity(self
     .cycle_time)))
322    axs[1].set(xlabel='time [s]', ylabel='velocity [  /s]')
323    plt.show()
324
325  def plot_knee_cycle(self):
326    fig, axs = plt.subplots(2)
327    axs[0].plot(self.cycle_time, np.rad2deg(self.get_knee_angle(self.
     cycle_time)))
328    axs[0].set(xlabel='time [s]', ylabel='angle [  ]', title='Knee
     trajectory')
329    axs[1].plot(self.cycle_time, np.rad2deg(self.get_knee_velocity(
     self.cycle_time)))
330    axs[1].set(xlabel='time [s]', ylabel='velocity [  /s]')
331    plt.show()
332
333
334
335  def plot_torque_cycle(self):
336    plt.plot(self.cycle_time, self.get_torque_sin(self.cycle_time))
337    axs[0].set(xlabel='time [s]', ylabel='angle [  ]', title='Knee
     trajectory')
338    axs[1].plot(self.cycle_time, np.rad2deg(self.get_knee_velocity(
     self.cycle_time)))
339    axs[1].set(xlabel='time [s]', ylabel='velocity [  /s]')
340    plt.show()
341
342
343  def plot_torque_step(self):
344    vec = [self.get_torque_step(t, self.step_perc) for t in self.
     cycle_time]
345    plt.plot(self.cycle_time, vec)
```

```
346     #axs[0].set(xlabel='time [s]', ylabel='angle [  ]', title='Knee
      trajectory')
347     #axs[1].plot(self.cycle_time, np.rad2deg(self.get_knee_velocity(
      self.cycle_time)))
348     #axs[1].set(xlabel='time [s]', ylabel='velocity [  /s]')
349     plt.show()
350
351
352   def stop(self):
353     self.driver.serial_write(0.0)
354     print('Motor stopped')
355
356   def run(self):
357     vel = self.get_knee_velocity(self.get_cycle_time())
358     vel_red = vel*self.gear_ratio
359     ref = self.get_torque_sin(self.get_cycle_time())
360     #ref = self.get_torque_step(self.get_cycle_time(), self.step_perc)
361     vel_red = vel*20
362     #print(vel_red)
363     #if ref != self.pre:
364     self.driver.serial_write(ref)
365     self.pre = ref
366     value = self.driver.serial_read()
367     self.read_array.append(value)
368     self.write_array.append(ref)
369     self.increment_time()
370
371
```

Appendix D

Knee tests results

D.1 Zero input control

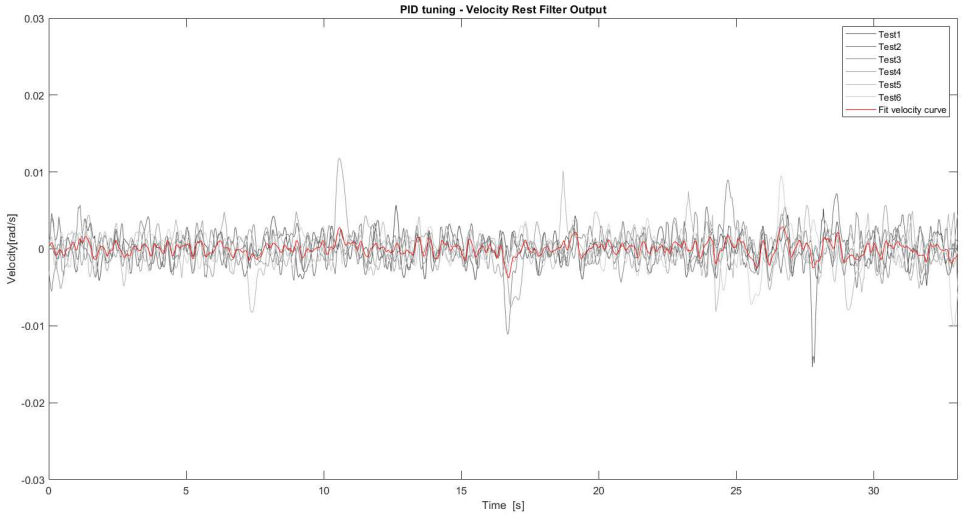
D.1.1 Velocity control ring enabled

D.1.2 Position control ring enabled

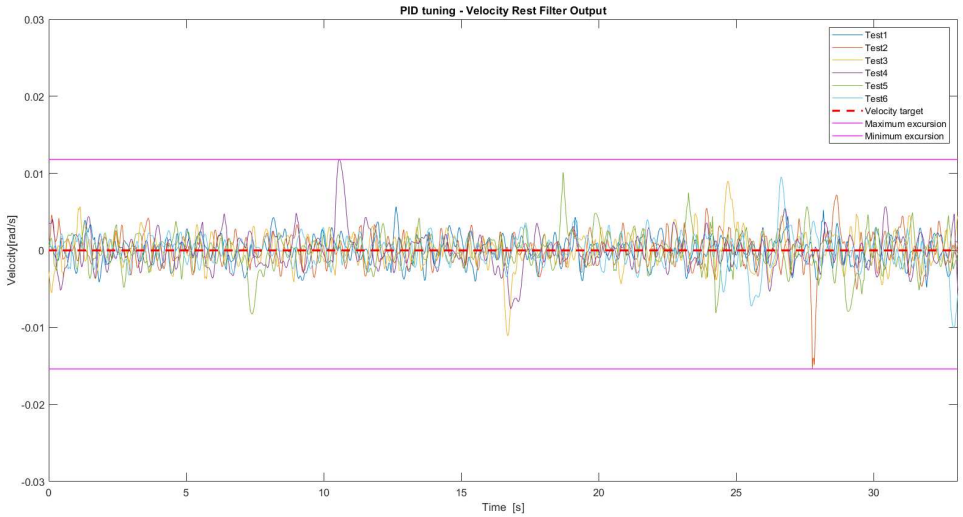
D.2 Step input contro

D.2.1 Velocity control ring enabled

D.2.2 Position control ring enabled

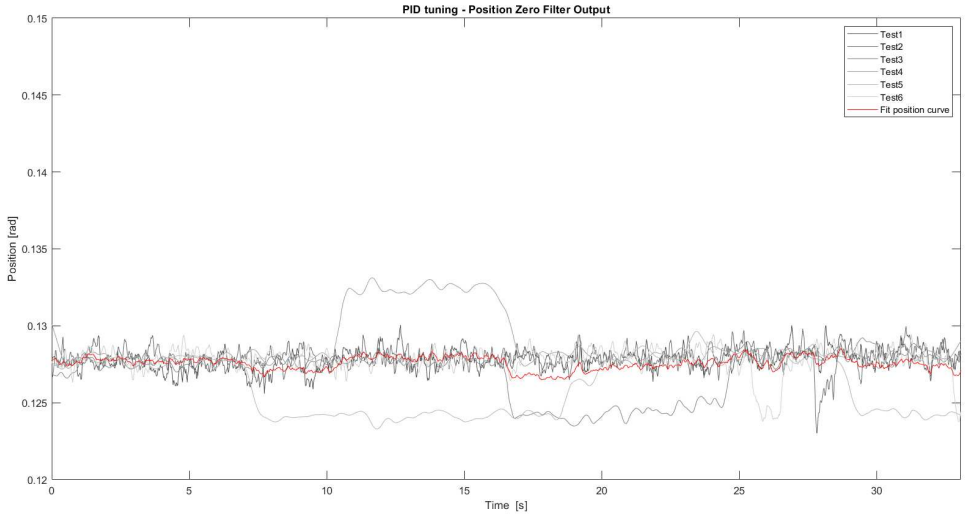

PID tuning - Velocity Rest Filter Output
Test1
Test2
Test3
Test4
Test5
Test6
Fit velocity curve
Velocity[rad/s]
Time [s]

(A)

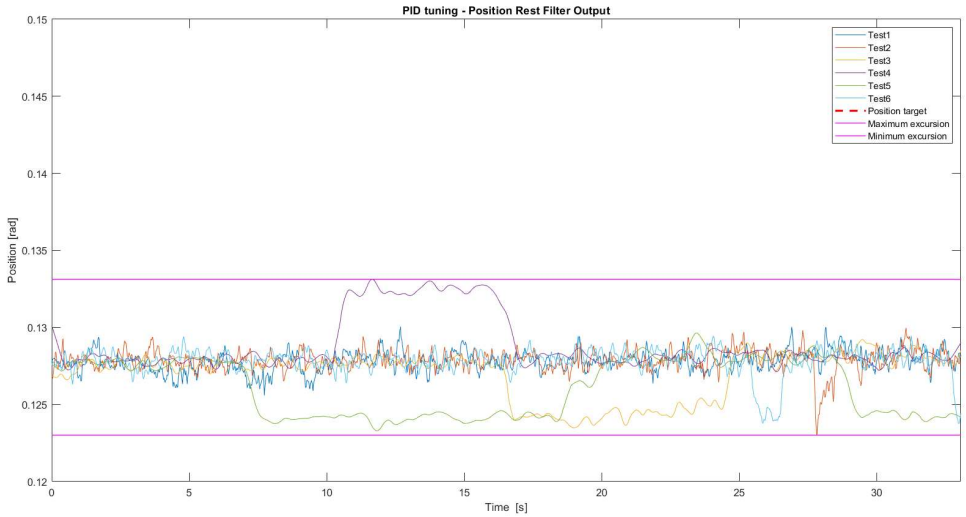

PID tuning - Velocity Rest Filter Output
Test1
Test2
Test3
Test4
Test5
Test6
Velocity target
Maximum excursion
Minimum excursion
Velocity[rad/s]
Time [s]

(B)

FIGURE D.1

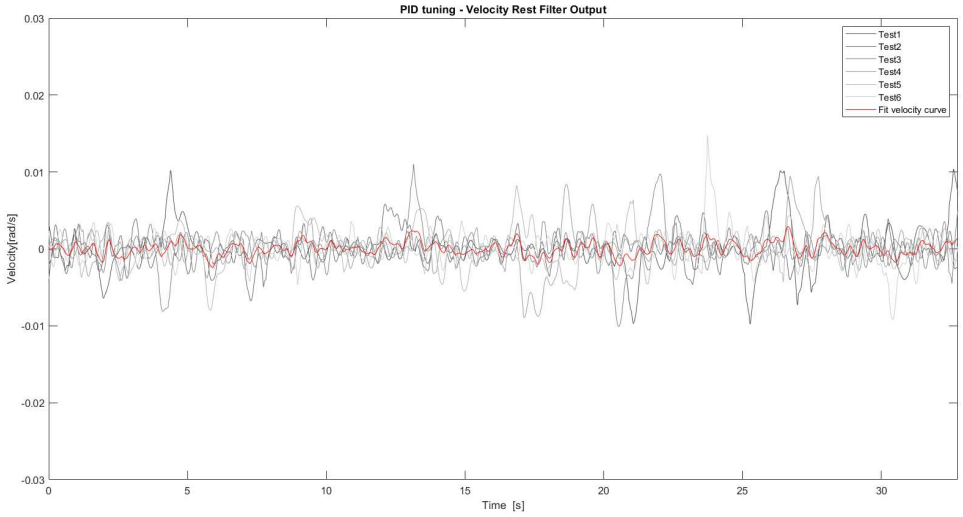

PID tuning - Position Zero Filter Output
Position [rad]
Time [s]
Test1
Test2
Test3
Test4
Test5
Test6
Fit position curve

(A)

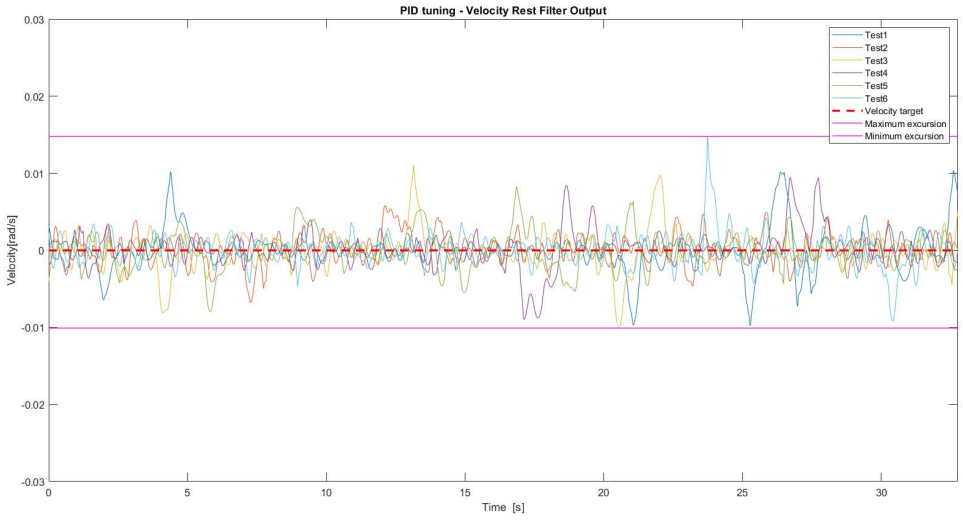

PID tuning - Position Rest Filter Output
Position [rad]
Time [s]
Test1
Test2
Test3
Test4
Test5
Test6
Position target
Maximum excursion
Minimum excursion

(B)

FIGURE D.2

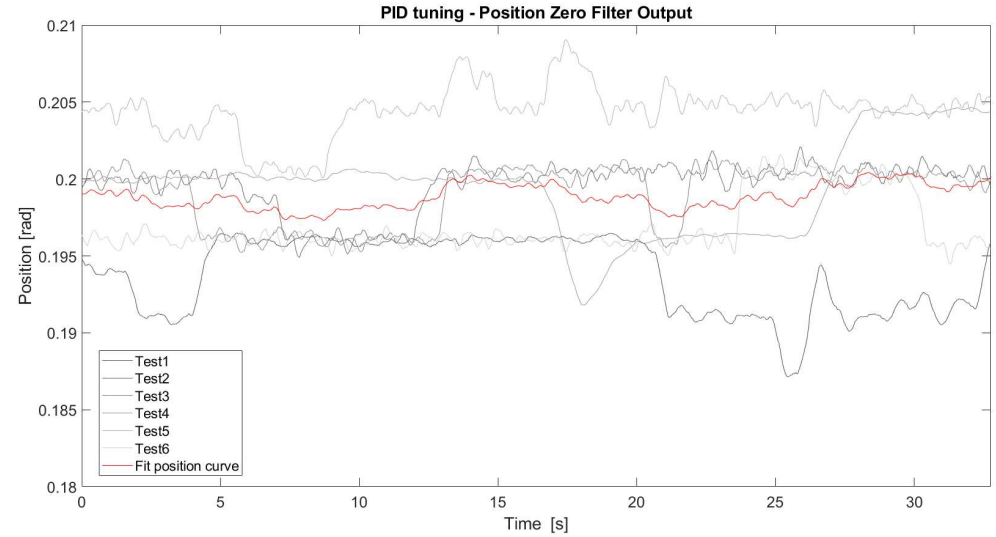

PID tuning - Velocity Rest Filter Output
Test1
Test2
Test3
Test4
Test5
Test6
Fit velocity curve
Velocity[rad/s]
Time [s]

(A)

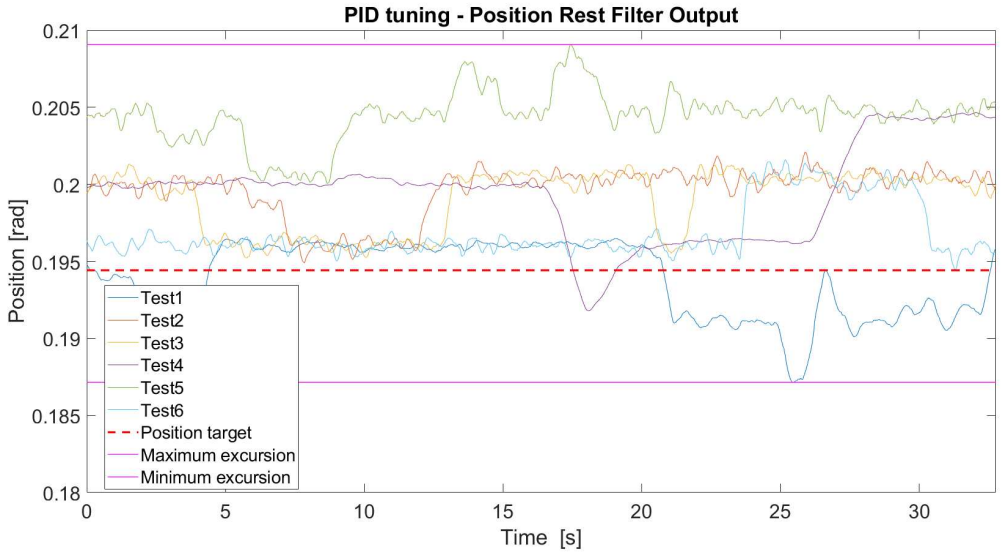

PID tuning - Velocity Rest Filter Output
Test1
Test2
Test3
Test4
Test5
Test6
Velocity target
Maximum excursion
Minimum excursion
Velocity[rad/s]
Time [s]

(B)

FIGURE D.3

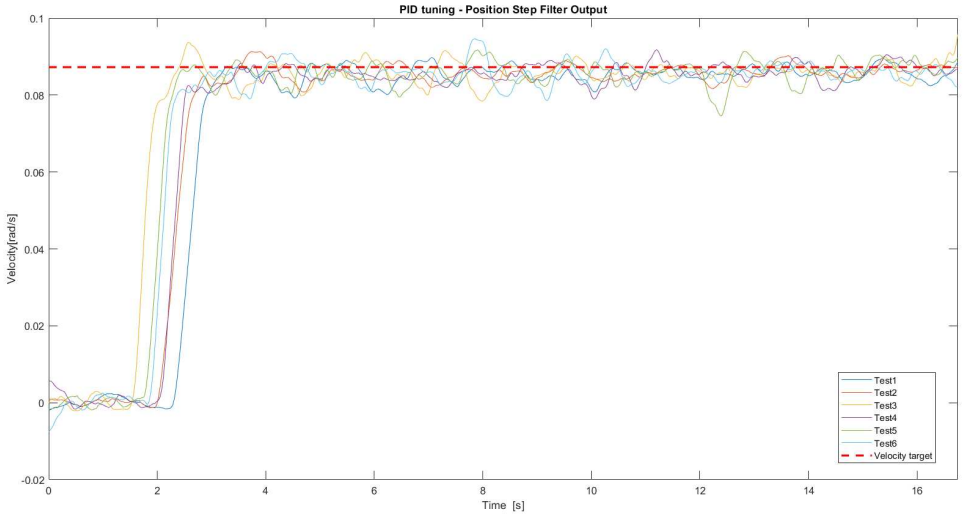

PID tuning - Position Zero Filter Output
Position [rad]
Time [s]
Test1
Test2
Test3
Test4
Test5
Test6
Fit position curve

(A)

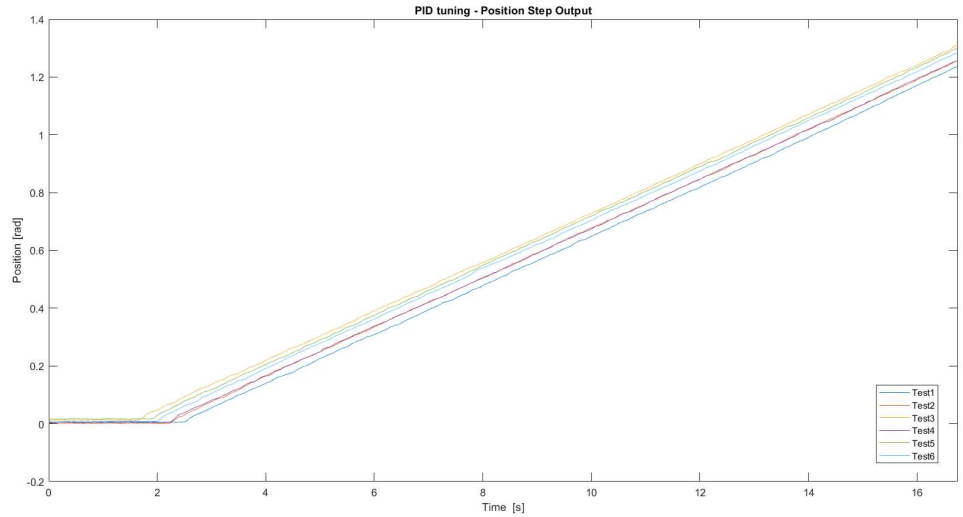

PID tuning - Position Rest Filter Output
Position [rad]
Time [s]
Test1
Test2
Test3
Test4
Test5
Test6
Position target
Maximum excursion
Minimum excursion

(B)

FIGURE D.4

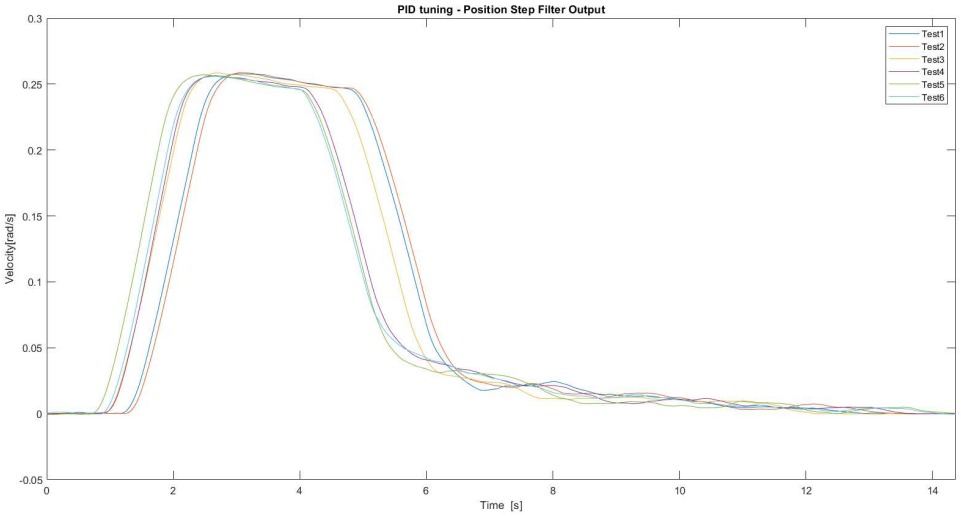

PID tuning - Position Step Filter Output
Velocity[rad/s]
Time [s]
Test1
Test2
Test3
Test4
Test5
Test6
Velocity target

(A)

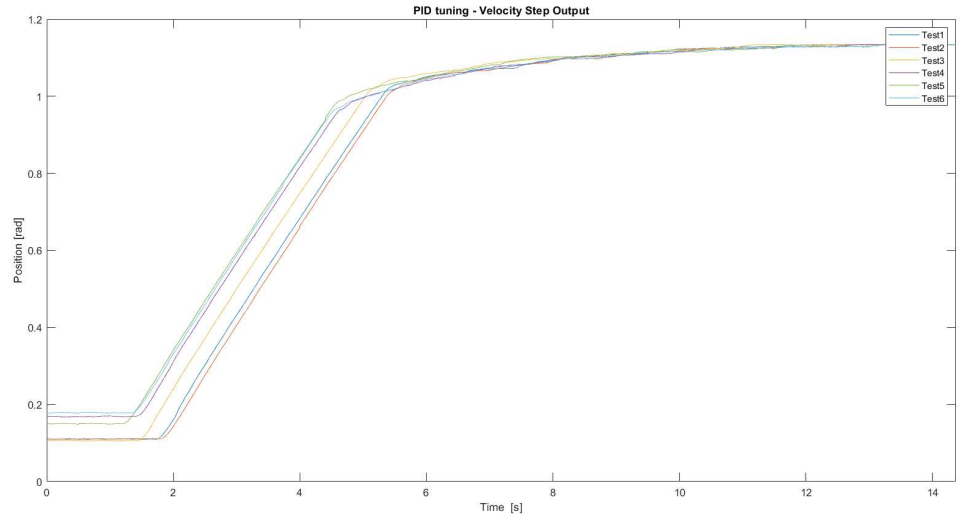

PID tuning - Position Step Output
Position [rad]
Time [s]
Test1
Test2
Test3
Test4
Test5
Test6

(B)

FIGURE D.5


PID tuning - Position Step Filter Output
Velocity[rad/s]
Time [s]
Test1
Test2
Test3
Test4
Test5
Test6

(A)


PID tuning - Velocity Step Output
Position [rad]
Time [s]
Test1
Test2
Test3
Test4
Test5
Test6

(B)

FIGURE D.6

Bibliography

Alberto Plaza Mar Hernandez, Gonzalo Puyuelo (2020). “Lower-limb medical and rehabilitation exoskeletons: A review of the current designs”. In: *IEEE Reviews in Biomedical Engineering*. URL: DOI10.1109/RBME.2021.3078001.

Alessio Martinelli Simone Morosi, Enrico Del Re (2015). “Daily Movement Recognition for Dead Reckoning”. In: URL: https://www.researchgate.net/publication/300443689.

Ali Nasr, Arash Hashemi and John McPhee (2022). “Model-Based Mid-Level Regulation for Assist-As-Needed Hierarchical Control of Wearable Robots: A Computational Study of Human-Robot Adaptation”. In: *robotics*. URL: https://doi.org/10.3390/robotics11010020.

Antonio Rodríguez-Fernández, Joan Lobo-Prat and Josep M. Font-Llagunes (2021). “Systematic review on wearable lower-limb exoskeletons for gait training in neuromuscular impairments”. In: *J NeuroEngineering Rehabil*. URL: https://doi.org/10.1186/s12984-021-00815-5.

Antun Skuric Hasan Sinan Bank, Richard Unger (2022). “SimpleFOC: A Field Oriented Control (FOC) Library for Controlling Brushless Direct Current (BLDC) and Stepper Motors”. In: *Journal of Open Source Software*. URL: DOI:10.21105/joss.04232.

Bhaben Kalita Jyotindra Narayan, Santosha Kumar Dwivedy (2020). “Development of Active Lower Limb Robotic-Based Orthosis and Exoskeleton Devices: A Systematic Review”. In: *International Journal of Social Robotics*. URL: https://doi.org/10.1007/s12369-020-00662-9.

Blaya, Joaquin A. (2005). “Force-Controllable Ankle Foot Orthosis (AFO) to Assist Drop Foot Gait”. In: URL: https://www.researchgate.net/publication/37990578.

Christopher Siviy Lauren M. Baker, Brendan T. Quinlivan (2022). “Opportunities and challenges in the development of exoskeletons for locomotor assistance”. In: *nature biomedical engineering*. URL: https://doi.org/10.1038/s41551-022-00984-1.

František Vaverka Milan Elfmark, Zdeněk Svoboda (2015). “System of gait analysis based on ground reaction force assessment”. In: *Acta Gymnica*. URL: doi:10.5507/ag.2015.022.

Grazia Cicirelli Donato Impedovo, Senior Member (ANUARY 2022). “Human Gait Analysis in Neurodegenerative Diseases: A Review”. In: *IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS* VOL. 26, NO. 1, URL: DigitalObjectIdentifier10.1109/JBHI.2021.3092875.

HE Guisong HUANG Xuegong, LI Feng (2022). “Review of Power-Assisted Lower Limb Exoskeleton Robot”. In: *J. Shanghai Jiao Tong Univ. (Sci.)* URL: https://doi.org/10.1007/s12204-022-2489-3.

Ioannis Papavasileioua Wenlong Zhangb, Song Han (2017). “Real-time data-driven gait phase detection using ground contact force measurements: Algorithms, platform design and performance”. In: *Smart Health*. URL: http://dx.doi.org/10.1016/j.smhl.2017.03.001.

J. C. González-Islas O. A. Domínguez-Ramírez, O. López-Ortega (2020). “Biped Gait Analysis based on Forward Kinematics Modeling using Quaternions Algebra”. In: *REVISTA MEXICANA DE INGENIERÍA BIOMÉDICA*. URL: dx.doi.org/10.17488/RMIB.41.3.4.

J.L. Pons, J.C. Moreno (2013). “Principles of human locomotion: a review”. In: *Annual International Conference of the IEEE EMBS*.

Joanne O Crawford Richard Graveling, Alice Davis (2020). “Review of research, policy and practice on prevention of work-related musculoskeletal disorders (MSDs)”. In: *European Agency for Safety and Health at Work* ISBN: 978-92-9479-357-7. URL: doi:10.2802/118327.

Juneja P Munjal A, Hubbard JB. (2024). “Anatomy, Joints”. In: *Treasure Island (FL) StatPearls*.

L. Wills S. Kannan, B. Heck (2000). “An open software infrastructure for reconfigurable control systems”. In: URL: DOI:10.1109/ACC.2000.878721.

Meby Mathew Mervin Joe Thomas, M.G. Navaneeth (2022). “A systematic review of technological advancements in signal sensing, actuation, control and training methods in robotic exoskeletons for rehabilitation”. In: *Emerald Insight*. URL: DOI10.1108/IR-09-2022-0239.

Monica Tiboni Alberto Borboni, Fabien Vérité (2022). “Sensors and Actuation Technologies in Exoskeletons: A Review”. In: *Sensors: Academic Editor: Helmut Karl Lackner*. URL: https://doi.org/10.3390/s22030884.

Robert M. Kanko Elise Laende, W. Scott Selbie (2021). “Inter-session repeatability of markerless motion capture gait kinematics”. In: *Journal of Biomechanics*. URL: https://doi.org/10.1016/j.jbiomech.2021.110422.

Tanner Amundsen Matthew Rossman, Ishfaq Ahmad (2022). “Fall risk assessment and visualization through gait analysis”. In: *Smart Health 25*. URL: https://doi.org/10.1016/j.smhl.2022.100284.

Tortora, Stefano and Alberto Gottardi (2022). “Shared Control in Robot Teleoperation With Improved Potential Fields”. In: *IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS*. URL: https://doi.org/10.1109/THMS.2022.3155716.

Tortora, Stefano and Emanuele Menegatti (2022). “) ROS-Neuro: An Open-Source Platform for Neurorobotics.” In: *Front. Neurorobots*. URL: doi:10.3389/fnbot.2022.886050.

Weiguang Huo Samer Mohammed, Juan C. Moreno (2016). “Lower Limb Wearable Robots for Assistance and Rehabilitation: A State of the Art”. In: *IEEE SYSTEMS JOURNAL* , VOL. 10, NO. 3. URL: DigitalObjectIdentifier10.1109/JSYST.2014.2351491.

Wills, Kannan and Vachtsevanos (2000). “AN OPEN SOFTWARE INFRASTRUCTURE FOR RECONFIGURABLE CONTROL SYSTEMS”. In: *Georgia Institute of Technology, Atlanta*.

Yongjun Shi Wei Dong, Weiqi Lin and Yongzhuo Gao (2022). “Soft Wearable Robots: Development Status and Technical Challenges”. In: *Sensors*. URL: https://doi.org/10.3390/s22197584.

Young, Aaron J. and Daniel P. Ferris (2017). “State of the Art and Future Directions for Lower Limb Robotic Exoskeletons”. In: *IEEE TRANSACTIONS ON NEURAL SYSTEMS AND REHABILITATION ENGINEERING* , VOL. 25, NO. 2, URL: DigitalObjectIdentifier10.1109/TNSRE.2016.2521160.

Önen Fatih M. Botsalı, Mete Kalyoncu Ümit (2017). “Design and Motion Control of a Lower Limb Robotic Exoskeleton”. In: URL: http://dx.doi.org/10.5772/67458.