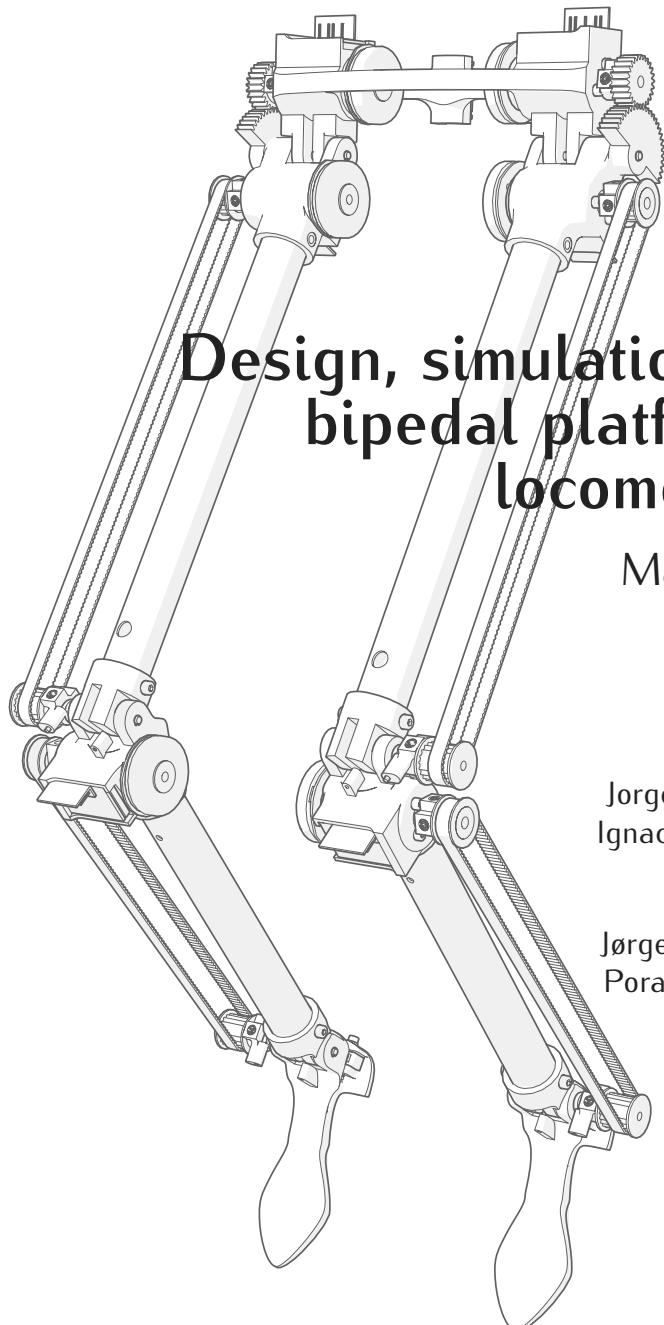


University of Southern Denmark
Faculty of Engineering
MSc in Engineering, Robot Systems



Design, simulation and construction of a bipedal platform for human-like locomotion studies

Master Thesis

Authors

Jorge Rodriguez Marin
Ignacio Torroba Balmori

Supervisors

Jørgen Christian Larsen
Poramate Manoonpong

May 31, 2016

*To Ignacio Torroba.
Jorge Rodriguez*

—

*To Jorge Rodriguez.
Ignacio Torroba*

Abstract

This is the abstractt

Resumen

Aquí va el resumen en español

Acknowledgements

These are the acknowledgements.

Contents

1	Introduction	1
1.1	Overall description	1
1.2	Goals	2
1.3	Project management	3
1.4	Report structure	5
2	State of the art	7
2.1	The evolution of bipedalism	7
2.1.1	Biped motion and engineering	8
2.2	From the wheel to the leg	10
2.2.1	Discovering the wheel (in robotics)	10
2.2.2	Achieving legged motion	11
2.3	Biomechanics meets Control	12
2.3.1	The relevance of body dynamics	13
2.3.2	Embodied AI and locomotion	14
2.3.3	The advancement of hardware	15
2.4	The next generation of bipeds	16
3	Conception and initial analysis	19
3.1	Bipodal locomotion	19
3.1.1	Walking and running gaits comparison	19

3.2	Geometrical dimensions of the frame	20
3.2.1	Limbs length	20
3.2.2	Hip and sole width	22
3.3	Inertial dimensions of the frame	22
3.3.1	Mass of the links	22
3.3.2	Mass distributions	23
3.3.3	Moments of inertia	23
3.4	Joints design	23
3.4.1	Actuators	24
3.4.2	Torque transmission	24
3.4.3	Joints compliance	25
4	Mathematical model of RuBi	29
4.1	Actuators design	29
4.2	Analysis of vertical jump	30
4.3	Kinematic model	33
4.4	Dynamic model	35
4.5	Springs influence in the actuation	37
4.5.1	Torque and energy contribution of passive actuators	37
4.5.2	Joint kinetics	37
4.6	Model-based controllers	38
5	Mechatronic design	41
5.1	Electronics	41
5.1.1	The electric actuators	41
5.1.2	Suitability of the motor model for the application	42
5.1.3	Embedded electronics	45
5.1.4	Sensory feedback	45

5.2 Mechanics	46
5.2.1 Pulleys and belts	46
5.2.2 Gears	48
5.2.3 Landing impact force	48
5.2.4 Limbs profile	50
5.2.5 Ankle and knee joint mechanics	52
5.2.6 Finite Element Method (FEM)	57
5.2.7 Computer-Aided Design (CAD)	57
5.2.8 Mechanical limits of the joints	59
5.3 Software	59
5.3.1 Motivation	59
5.3.2 ROS Control	61
5.3.3 ROS Control-to-Hardware interface	62
5.3.4 Example controllers	63
6 Simulation	67
6.1 Motivation	67
6.2 Comparison of simulators	68
6.3 Robot definition	69
6.4 Robot creation	69
6.5 Dimensional analysis	71
6.6 Environment creation	73
7 Construction process	75
7.1 3D printing	75
7.1.1 Arc compensation in FFF	76
7.2 Machining	78
7.3 PCBs and wiring	78

7.4 Providers	78
7.4.1 Components delivery	79
7.5 Assembly	80
8 Experimental framework	81
8.1 The test bench	81
8.2 Vertical jump experimentation	81
8.3 Simulation experiments	84
9 Economical analysis	85
10 Results	87
10.1 Morphological properties of RuBi	87
10.2 Actuation and sensory feedback	88
10.3 RuBi dynamic model and actuators calculation	89
10.4 Framework for experimentation	90
10.5 Simulation environment	90
10.6 Mechanical implementation	91
11 Discussion	93
11.1 The first prototype of RuBi	93
11.2 The simulation and control architectures	93
11.3 The mathematical framework	94
11.4 The test bench for experimentation	94
12 Conclusions and further work	95
12.1 Further work	95
Appendices	101
.1 Profile selection	104

.2	Mechanical drawings	105
.3	CAD parameters	112
.4	Available springs	113
.5	Gears of the hip	113

List of Tables

3.1	Human proportions from DIN 33402-2.	20
3.2	Limbs index.	21
5.1	Internal register addresses of motor boards.	45
5.2	Input parameters for calculating the impact force.	49
5.3	Stress and deformation analysis for the sections given.	52
5.4	Profile selection for each limb.	52
9.1	Material cost.	86
10.1	Physical properties of the limbs obtained from CAD program.	89
1	Springs models (left configuration)	113

List of Figures

1.1	Screen capture of Trello showing the 1st sprint board. The three rightmost lists are used to keep track of tasks and their state.	3
2.1	Evolution of bipedalism (artistic depiction from [71]).	8
2.3	Examples of wheeled robot platform.	11
2.4	Examples of legged robot platforms.	13
2.5	Examples of bio inspired mechanics in locomotion.	14
2.6	Examples of humanoid robots.	16
3.1	Lower body measurements used for RuBi. Picture adapted from [78].	21
3.2	Four configurations for elastic actuators as from [43].	26
3.3	SEA configurations studied.	27
3.4	PEA configurations studied.	27
4.1	Jump phases without fall and landing.	31
4.2	Force-application time values for the same work magnitude.	33
4.3	Coordinate definitions for the schematic model of one leg.	34
4.4	Joints trajectories from kinematics model.	39
4.5	Joint torque and velocity as a function of time.	39
5.1	Left leg extension PCB schematic.	43
5.2	Results of Algorithm 1 for the given inputs (not all the $(\theta_{i,j}, \tau_{i,j})$ pairs are plotted).	44
5.3	Teeth detail from pinion and gear.	48

5.4 Impact force decomposed.	50
5.5 Simplified representation of the bending case.	51
5.6 Knee joint mechanical implementation.	53
5.7 Transversal section of the parallel spring resting positions in the left ankle.	56
5.9 Main CAD designed components	59
5.10 Figures of the joint limits implementation in RuBi.	60
5.11 ROS Control and Gazebo data flow.	61
5.12 RQT workspace containing the CPG signals and the dynamic reconfigurable parameters.	64
6.1 Simulation examples of the software analyzed.	68
6.2 Gazebo showing physical and visual properties of RuBi.	70
6.3 Coordinate system of the left ankle in SolidWorks.	71
6.4 Representation of a minimal volumetric cube.	72
6.5 Rotational robot holder and robot.	73
7.1 Gcode visualization from Simplify3D [62].	76
7.2 Photos of 3D printed parts.	77
7.3 Technical representation of the generated arc when using FFF technology.	77
7.4 Final architecture of electronics.	79
7.5 Actual photos of RuBi.	80
8.1 Photo and render of the test bench deployed.	82
8.2 Virtual spring representation in different states.	83
8.3 Screenshots of the robot taken when jumping by using the impulse controller.	84
10.1 Length-weight comparative with existing bipeds.	88
10.2 Measurements obtained from real robot.	88
10.3 Snapshots of RuBi performing an unconstrained jump.	90

Chapter 1

Introduction

1.1 Overall description

The present thesis aims at laying the foundations of the construction of a robust, low-cost and reconfigurable bipedal locomotion platform for human-like walking and running gait generation and control studies. The idea behind this project is to contribute to the on-going line of research in locomotion control at the AI department at the Maersk Mc-Kinney Møller Institute with a new biped robot for further investigation.

The devise, construction and testing of a functional robot of this complexity is by definition a task of great magnitude that involves the conjunction of several disciplines within the robotics field, including electromechanics, structural design, materials science and computer science and control, among others. Such a complex and extended work, together with the time constraints imposed, have made the authors of the project conceive its development with the view in a further growth beyond the scope of a single master thesis. With this idea in mind, one of the goals aimed here was to carry out the project facilitating to the farthest possible extent the continuity of its development by new students and researchers. Hence, both the hardware and software tools utilized and developed have been thought to be as general-purpose as possible, and all the necessary information required to replicate the work performed has been carefully documented here.

With the above explained as guidelines for the whole process the mechanical design of the robot has been carried out striving for simplicity and functionality, for which 3D-print manufacturing has been applied in order to speed up the prototyping, reduce costs and ease the testing. The conception of the structure is based on standard human parameters with the purpose of mimicking the dynamics of human legs and approximating the final results as much as possible to the gaits performed by people. The existing software for simulation and control of bipedal robots has been migrated to ROS [72] and Gazebo [32] to give a bigger portability and generality to the application, and the developed controllers and interfaces are ROS-based as well. The electronics hardware utilized in this thesis consists mainly in components developed for the Locokit project [41] created at the University of Southern Denmark, which have been adapted to the specific requirements of the task. Furthermore, in an early stage of the project the platform was meant to be actuated by an existing,

although extended, neural control and therefore it has been designed in accordance to its characteristics.

1.2 Goals

The objectives of the research and mechatronics work this thesis has implied are detailed below. Due to the ambitious scope of the project and the wide range of tasks involved it was hard to assess at the initial proposal the time needed and therefore the main goals of the thesis.

The Running Bipedal robot RuBi

The robot RuBi has been devised as a platform to be used for testing new control algorithms used for the generation of human-like walking and running gaits and the transition between them. Therefore, it has to count on a reliable hardware structure able to offer all the needed functionalities, together with a robust software frame capable of dealing with the requirements of such algorithms¹.

RuBi as a study framework

This thesis aimed at creating not only the robot described below, but a whole framework for legged locomotion studies. To do so, some more tools besides a first prototype of the robot were targeted as goals, and are listed in 1.2. Some of them have emerged from the actual process of design and construction of the robot, and others have been included with the view on its future use for research.

- A more general simulation environment than the existing one, based on ROS and Gazebo and eliminating dependencies with LPZ robots, together with a fully-scalable simulation model of the robot. Furthermore, the implementation of the ROS Control set of packages [58] in the system allows for a simple implementation of controllers that can be easily interfaces with Gazebo or the actual robot. This has been tested with two custom controllers.
- A test bench designed to hold the robot limiting its movements to the motion plane under study and guaranteeing its stability over a treadmill for locomotion experimentation.
- A mathematical framework to be used for computation of both kinematic and dynamic parameters and potentially for developing classical controllers from the dynamic model of the robot.
- An easy to modify and extend software interface with the electronic hardware, aimed at providing a simple and functional control of the robot.

¹The capabilities that walking and running gaits generation requires are detailed in 3.1.

Even though some of these goals have not been fully accomplished in the sense that they still need some improvement to be fully functional, as explained before all the necessary steps to ease the continuity of their development have been taken. Furthermore, in the original description of the project was also stated that, if possible, some experiments would be devised and carried out in order to test the functionality and physical properties of the prototype. However, some eventualities detailed in the next chapters have prevented the project from reaching that stage and a more complex experimentation phase besides the simple tests of functionality are left as further work.

1.3 Project management

The methodology used for this project is Agile. Agile approaches help teams respond to unpredictability through incremental, iterative work cadences, known as sprints. It has been implemented as SCRUM [63] by considering both supervisors as the product managers and having a self managed team striving to build properly tested product increments within short iterations. By having weekly meetings, empirical feedback has been received closing the iteration circle faster and more concisely.

The tool Trello [73] has been used in order to keep track of the tasks in the backlog. Trello is an online project management application and it was used since its design made it similar to post-it notes for management. It also has the advantage of being online and therefore accessible from both home and workplace through smart phones or computers. Figure 1.1 shows a screen capture of Trello board for a sprint log.

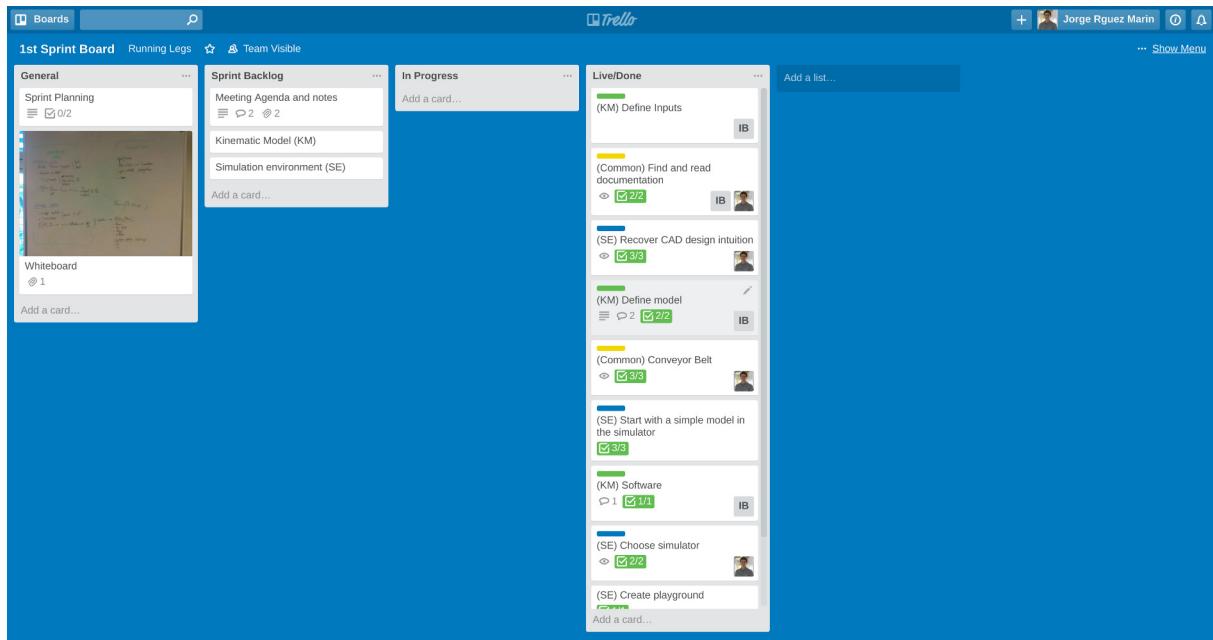


Figure 1.1: Screen capture of Trello showing the 1st sprint board. The three rightmost lists are used to keep track of tasks and their state.

Besides, the project has been organized in four sprints with big tasks that then are broken down into smaller jobs. The sprints are:

Sprint 1

- **Goal:** Analyze robot possibilities, define the kinematic model, become familiar with the simulation environment and get ready ROS Control and software.
- **Duration:** 21st of January to 28th of February.
- **Description:** An analysis of the structural possibilities of the robots is done. This is then used to start the mathematical model, which is subdivided in the kinematic and the dynamic model so the first one can be started. In parallel, all the software environment is created: from ROS control to the Gazebo structures. This will help with future extensions and will facilitate its integration for new features like the real CAD model of the robot.

Sprint 2

- **Goal:** Define the dynamic model, integrate Go Robots in the software structures and prepare simple simulation for testing and create example controllers.
- **Duration:** 3rd of March to 1st of April.
- **Description:** The kinematic model is finished and the dynamic starts. From the software point of view, a basic simulation model is created and controlled with ROS Control. The example controllers, based on tools from Go Robots, are created and tested.

Sprint 3

- **Goal:** Obtain data from the mathematical model, design the CAD model of the robot and obtain the final order list.
- **Duration:** 2nd of April to 1st of May.
- **Description:** The mathematical model is finished and it is used to size the motors and other components, that lets start the electro-mechanical design of the robot. If not completely finished, a list of the parts to be obtained is done and ordered.

Sprint 4

- **Goal:** Make electronics, print the parts with the 3D printers, receive ordered components, add software interfaces, assemble the robot, carry out tests and write thesis.
- **Duration:** 1st of May to 31st of May.
- **Description:** While getting the parts, others like the 3D printed or machined parts are created. The electronic interfaces are also made. From the software perspective the interface with the real robot is implemented. The robot is assembled and the experiments carried out. Finally, the documentation of the project starts.

1.4 Report structure

The document is organized so it starts with an intensive analysis of the current technologies and solutions in **State of the art** [2]; continues with an analysis of the problem and design boundaries are established in the **Conception and initial analysis** chapter [3]; in the **Mathematical model of RuBi** [4] a generalized kinematic and dynamic model of a biped is got that is then used to size the actuators; then, in the **Mechatronic design** chapter [5], divided in *mechanics* [5.2], *Electronics* [5.1] and *Software* [5.3], the integral and holistic design of the robot is carried out; the **Simulation** chapter [6] is meant to explain why and how the simulation tools where included in the framework; it follows with an **Construction process** chapter [7] where processes as the manufacturing, the wiring or the election of the providers is explained; in the **Experimental framework** chapter [8] are depicted the performed tests and the suggested ones, whilst in **Economical aspects** [9] the budget of the project is specified; it finishes with the chapters **Results** [10], **Discussion** [11] and **Conclusions and further work** [12].

Chapter 2

State of the art

This chapter contains a concise overview of some of the major, and more relevant for this thesis, advances in artificial legged locomotion in the past 60 years. A brief prelude of the emergence and evolution of natural, limb-based terrestrial motion and more in detail bipedalism is presented. It is followed by a short introduction of some of the history and results from the union between the engineering field and the study of motion and legged systems. From the first, ancient mobile artifacts and prosthesis to the current, most advanced devices in robotics. It continues with an exposition of three of the most pertinent breakthroughs in modern technologies that allowed to reach the current state of the art in modern autonomous walking robots. And it finishes with a quick look to the future of the bipedal locomotion.

2.1 The evolution of bipedalism

The current human bipedal locomotion arises from the combination of a wide variety of subsystems working in conjunction to achieve the intended gait generation according to the requirements of the situation. The biomechanics of the limbs, consisting in bones, muscles and tendons under the control of the nervous system yields the adequate production of the different motion patterns in order to displace the body as energetically efficiently as possible. This complex behavior is the result of 4 million years of an evolution [60] that started in primates and that has entailed both morphological and neurological changes in the human body since the first bipedal hominids to the current structure in *Homo sapiens*. There exist several different theories about the reasons that originated and led to the adaption of this posture and bipedal motion. Although different, most of them assume that most of them assume that the development of this structural and behavioral changes arose from a change in the environment in pre-hominids, for with bipedal behavior suddenly offered some kind of survival value [1]. However, the genus *Homo* is not the only species that has evolved towards two-legged locomotion. Currently there are a few more species that have reached this method of displacement as a result of a natural selection process in which bipedalism offered the broadest set of advantages of the specie being the main ones listed below.

- Erect posture for a wider field of view and reach range.
- Free forelimbs, that could evolve towards specialized, non-locomotory applications such as object manipulation, combat, flight, etc.
- Faster displacement in certain species, although not generally.

Extensive research in the actuation and control structures involved in human bipedalism has been conducted from within the scientific fields of anthropology, biology, medicine, sport science and lately, several areas within the engineering. Its goal, as per definition of science, has been to reach a full understanding of what led to this behavior and the knowledge of how it functions, together with the discovery of ways in which it can be mimicked and improved, aiming at a more inexpensive and optimized locomotion. This last fact has led to the belief that the next stage in the evolution of human locomotion will not come from nature as until nowadays, but from the hand of science and engineering, which has been lately depicted in literature and pop-culture as in 2.1.

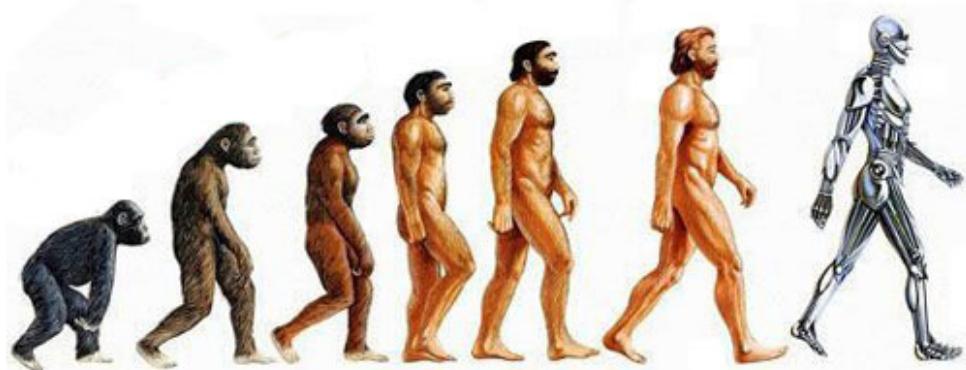


Figure 2.1: Evolution of bipedalism (artistic depiction from [71]).

2.1.1 Biped motion and engineering

The discipline of engineering has been historically among the latest of the cited ones to start its contribution to the research and improvement of biped motion in a well defined manner, although its earliest contributions seem to date from ancient Egypt and India [22]. From the study of human bipedal motion described above, the insights emerged regarding its biomechanical and control functioning together with the need to restore, improve or imitate its capabilities led to the creation of new branches within the discipline of engineering. The three most relevant ones for the present thesis are listed and introduced below.

1. Orthotics
2. Prosthetics
3. Artificial legged locomotion

Orthotics Orthotics is a specialty within the medical field concerned with the design, manufacture and application of orthoses. An orthosis is an externally applied device used to modify the structural and functional characteristics of the neuromuscular and skeletal system, as per definition in [54]. There are orthoses dated back 2000 years ago where layers of wool were placed inside the sandals to give relief to foot fatigue or strain. The cutting edge in this field make use of the new manufacturing technologies like 3D printing or 6 axis CNCs milling machines, which allow the creation of devices with complex shapes and novel materials. As introduced in [37], the most critical part of the orthosis is the socket, being this the interface between the limb and the device. Its comfortability has been improved by making use of new materials like sole-elastomers. Sockets made of different materials can be used to get different physical properties depending on the limb of the patient, providing user adaption. In the most advanced pole of this specialty, the powered exoskeletons can be found. Examples like [79], [76] or [53], show how these machines are meant to help increase the physical capabilities of a person, handicapped or not.

Prosthetics Prosthetics is the field of medicine that comprises the design and creation of prosthesis, defined as artificial limbs aimed at restoring motor and sensory capabilities in amputee patients. Since the first lower-limb prosthesis implant recorded in history, documented in the Rigveda [22], to the current state of the art there has been more than 3000 years of development. This time has taken prosthetics from single-piece, non-articulated devices with no actuation or sensory feedback to the current near-natural, anthropomorphic structures and control systems adapted to the patient needs and able to closely provide the properties of a biological limb. As in the orthoses design, the aim of the prosthetics is to mimic as closely as possible the human capabilities with designs such that they adapt to the subject as naturally as possible. Thus, their development goes parallel to the research and understanding of the human body. One of the most comprehensive studies in lower-limb prostheses, their design and actuation is the one provided in [43].



(a) Prosthetic leg, Össur Flex-Run. (b) Knee orthosis, New Hope Co. (c) Robotic legs, COMAN robot [74].

Artificial legged locomotion The first artificial walking artifacts recorded date from the ancient Greece. It is in this period when humankind first tries to imitate and replicate the structures found in nature, creating mechanisms to get a deeper knowledge of their

functioning and mimic them. But it is during the Renaissance in Europe when the developments in mechanics and the study of the nature and the human body allows to create the first automata able to walk as a predefined combination of complex mechanical operations. It is after the Second World War, with the application of electronics and computer technology that the advancement of walking machines is revolutionized [18].

The engineering branch of modern robotics has its origins in the second half of the 20th century has a conjunction of the areas of electrical and mechanical engineering and computer science. However, its interest in achieving resemblance to the locomotion means of both humans and animals and their capabilities arose in the 1980's. It first crystallized into the construction of Wabot-1 at WASEDA University or the creation of the Leg Laboratory by Marc Raibert at the Massachusetts Technological Institute. The purpose of the research in this field was the construction of useful knowledge basis on how human and animal locomotion works that could be used in the creation of legged vehicles, as described in [15].

2.2 From the wheel to the leg

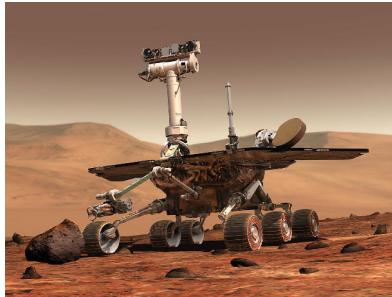
While the wheel is a young human invention meant to facilitate terrestrial motion, legged locomotion has been the result of millions of years of adaption of the species from the aquatic to a terrestrial field. During this time, even though other forms of motion on the ground emerged, including limbless or rolling, the tetrapod and quadrupedal motion are the ones that have proved the best performances in terrestrial displacements in terms of relative velocity or jump length, for instance.

2.2.1 Discovering the wheel (in robotics)

Despite the facts above, it was the wheel the mean chosen to implement the ability to move around on the first mobile robots such as Walter's tortoises or the John Hopkins University robot Beast [31], [39]. This could be explained due to the fact that, as an artificial human creation, the modeling and control of the wheel could be fully mastered during its evolution process, easing its implementation in robotic platforms. Some examples of this are shown in Figure 2.3, which depicts three current applications of the wheel in robot platforms.

Although the wheel was chosen as the motion mean to face what can be considered one of the most challenging and unpredictable environment a robot can be subject to, the surface of a new planet, it was not the only option. The Space General Corporation firstly studied the use of multi-legged autonomous platforms for lunar exploration in the early 1960's [18]. However, their use was eventually discarded due to its poor adaptability due to a too low number of DOF. The "Spirit" robot in 2.3a, within the NASA's Mars Exploration Rover mission, is the example of the current solution to extraplanetary exploration and a good example of the possibilities and limits entailed by wheel-based locomotion. It was successfully launched on Mars surface in 2004 and it stayed active and operative until 2010. However, the end of the mission came from the hand of the terrain. A very low-cohesion area of soil made the wheels of the device loose traction and eventually get

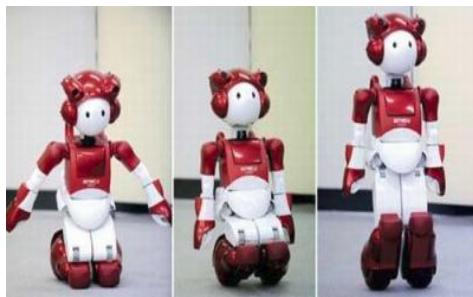
trapped, preventing from recovering the control of the vehicle and bringing about the end of the mission.



(a) Mars Rover Spirit by NASA.



(b) MIR 100 by MIR.



(c) EMIEW2, by Hitachi.

Figure 2.3: Examples of wheeled robot platform.

Figure 2.3b shows a MIR 100 robot, a state-of-the-art service robot designed for independent transportation and logistics in a dynamic and unpredictable environment.. As a service robot, it must fulfill strict safety standards and guarantee a secure interaction with users and other equipment while accomplishing its tasks. Thanks to its wheels configuration, it can achieve an agile mobility, fast reaction capacity and a big load capacity.

The EMIEW 2, in 2.3c, is an example of midpoint between wheeled and legged locomotion. Although its motion relies on steering wheels, their placement at the end of two leg-like lower-limbs provides with some of the advantages of bipedal motion, such as active suspension or impact absorption.

2.2.2 Achieving legged motion

In spite of the great successes attained in the field of wheeled-robotics, possible thanks to the development in engines and the increasing computing capacity of embedded processors, mobile robots based on wheels have not been able to achieve performances in motion

comparable to the ones found in nature. Specially in uneven terrains, velocity, maneuverability, and efficiency are still problems under study. Besides, the absence of wheels in nature as a result of darwinian evolution ultimately led to questioning [55] if they are truly the best mean to overcome the challenges that displacements in uneven, unknown surfaces yield for robots. In words of Richard Dawkins [55] "[the wheel] is dependent for maximum efficiency on a prior invention – the road (or other smooth, hard surface)". Figure 2.4 contains three of the most advanced stages in legged locomotion nowadays. They have been chosen because they are meaningful current paradigms of what brought about the revolution to the leg-based locomotion systems in the 80's: the conjunction between mechanics and control as the producers of the expected behavior [15].

The STARLeth robot in 2.4a, built at the Autonomous System Lab at ETH, is an example of robot legged robot inspired by nature. It is meant at proving that versatility, speed, robustness and efficiency can be achieved together in a legged moving platform [8]. But it also represents the control problems arisen from this kind of locomotion, by definition unstable and complex (12 actuated joints and 6 unactuated DOF).

In Figure 2.4b the Raptor robot created at KAIST is shown. This robot is inspired in a velociraptor structure and has been able to achieve a top stable speed on a treadmill of 46 km/h , making it the current fastest biped robot (although the Cheetah robot by Boston Dynamics has the record as a quadruped). However, it is discussed here because of its simple bioinspired design, which combines prosthetic blades with a dynamic tail for stabilization and only required two motors. This makes it a good illustration of the importance of biomechanics in the overall system.

Finally the ATLAS robot by Boston Dynamics, in 2.4c, represents the state of the art in bipedal autonomous platforms handling unknown, rough terrain and its interaction with dynamically changing environment. This task for a high degree of freedom mobile manipulator as a humanoid requires a very robust interactive motion planning and control in order to quickly carry out its assignment while overcoming unpredicted situations fast and reliably enough. The ATLAS is a good illustration of the relevance of a robust but flexible and adaptive control architecture for this kind of machines.

2.3 Biomechanics meets Control

The first attempts to accomplish artificial walking machines in the mid-1950s were based on stiff structures and kinematic control. There has been more than 60 years of development in several technology fields between them and the current soft, compliant and torque-controlled legged robots. Among the main advances that gave rise to the evolution of the autonomous walking robots, three considered essential and of great relevance for this thesis are listened in 2.3 and further discussed here. Thanks to them, nowadays the research in legged robotics goes hand in hand with the study of biological locomotion systems in the attempt to imitate existing features present in nature.

- The realization of the importance that a dynamic and active control has on the walking behavior, as opposed to rigid, kinematics-based motion.

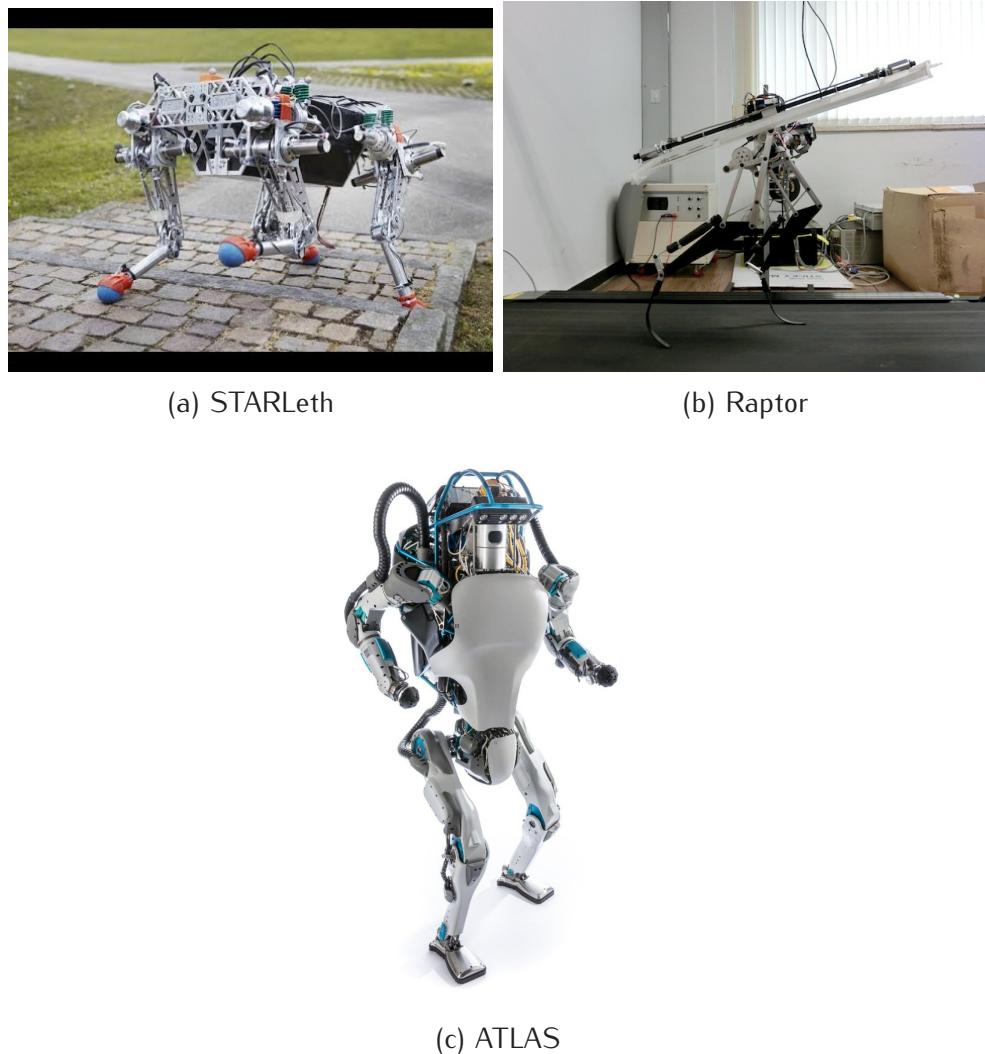


Figure 2.4: Examples of legged robot platforms.

- The conception of the embodied AI. The influence of the body in the process of thinking.
- The improvements in sensors/actuators performance, material science, embedded computing power and power sources.

2.3.1 The relevance of body dynamics

Stiff position control for kinematic trajectory planning has lately demonstrated impressive capabilities for instance during the last DARPA learning locomotion challenge. However, the limitations of this kind of control such as the need of a very detailed knowledge about both the robot state vector and the environment have been also exposed. Thus, it seems clear that the solution for robust and adaptive motion platforms has to go through the development of dynamics-based control models. The first big steps in dynamic legged locomotion control took place in the 90's in the Leg Laboratory, at the MIT Artificial Intelligence Laboratory carried out by Marc Raibert and his team. Their findings about the importance of active balance for stability and the possibility of creating simple and

generalizable control algorithms for complex dynamic legged systems [15] pushed the development of walking robots to a new stage. Besides, they were among the pioneers in the realization of the influence of the mechanical design together with the control in the generation of complex behaviors in locomotion. A paradigmatic example of the importance of the biomechanics and the body dynamics in the control of machines intended for locomotion is the well-known passive walker developed by McGeer [70] and shown in 2.5a. The achievement of a human-like, efficient walking behavior without any kind control structure gave birth to the concept of morphological computation.

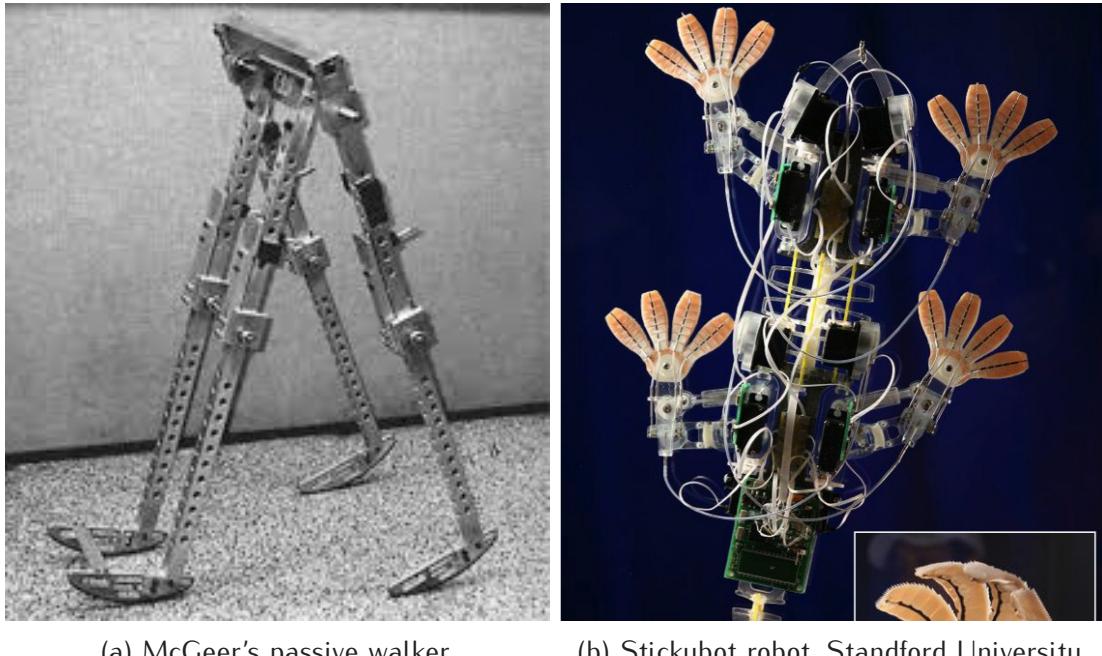


Figure 2.5: Examples of bio inspired mechanics in locomotion.

2.3.2 Embodied AI and locomotion

The embodiment concept came up in the Artificial Intelligence field in the 1980's as a mean to explore the influence of the body and its characteristics in the process of thinking. Its aim could be expressed as the search for the answer to the question, in the very convenient words of Rolf Pfeifer and Fumiya Iida, "How does walking relate to thinking?". As introduced in [13], the original goal of AI the understanding of natural forms of intelligence that have more to do with the interaction with the real world rather than the only development of computer algorithms.

The advent of this new approach in AI caused by itself an enormous boost in the field of locomotion in robotics since it made researchers start working with mobile robots. "The general initial conviction that locomotion and orientation were the underlying driving forces in the development of cognition, in the evolution of the brain, led to the general use of the already available wheeled robots" [13]. In the seek of an answer to the famous question "Why don't plants have brains?", by D. Wolpert, the belief that the reason could be their incapacity for displacement led to an increasing use of mobile robots as a research tool.

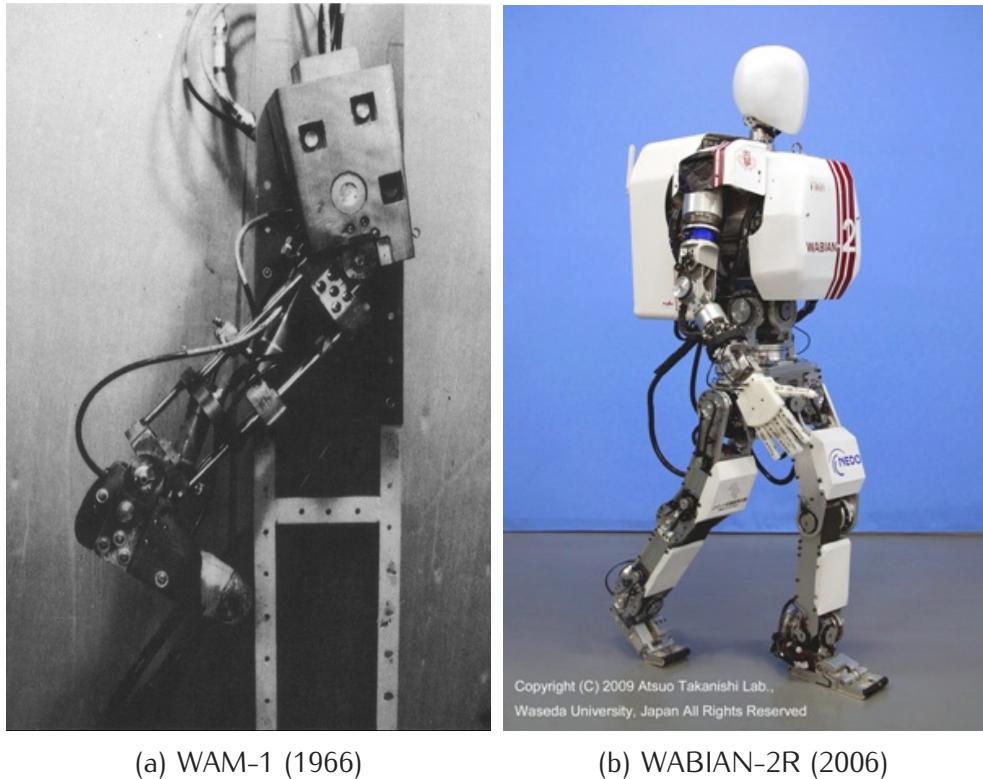
Besides, the embodied AI brought about the turn to nature for inspiration in biological systems under the belief that the results of Darwinian evolution and the principle of ecological balance had created systems of greater complexity and efficiency worth studying and mimicking. With all this, it came the understanding that complex behavior could arise from the synergistic combination of simple algorithms and the physical characteristics of the body. Since then, mechanical compliance in the actuation, soft materials and bio inspired controllers such as CPGs for instance have been widely studied and implemented in walking machines. An example of this concept is the Stickybot, created in the Biomimetics and Dexterous Manipulation Lab at Standford University and shown in Figure 2.5b.

2.3.3 The advancement of hardware

The early timeline of the evolution in the technology utilized in biped robotic systems can be analyzed taking a look at the progression of the humanoid robots at Waseda University in Japan, in the Humanoid Research Laboratory. In the late 1960's, Ichiro Kato and his team created the WAP-1, the first version of their first generation of humanoid robots, and the WAM-1 2.6a, matching the launching of the first microprocessors. The WAP-1 robot was moved by very slow pneumatic artificial muscles, first developed in the 1950's but not widely commercialized until the beginning of the 80's, and had to be connected to a large external computer frame for its control. Later version WAP-2 and WAP-3 incorporated controller-based memory and PWM-driven actuators, created at the end of the 1960's, that allowed to reach three-dimensional walking at near human-like locomotion speeds for the first time in history. Kato also created the first full-size humanoid robot, the Wabot-1, and since the beginning of the 80's his group and him applied for the first time 16-bit microcomputers (introduced in 1973) to the implementation of quasi-dynamic walking control [12]. One of the latest achievements of Kato Laboratory is the WABIAN-2R, in 2.6b.

The above is just an example of how during the last two decades of the 20th century and the beginning of the 21st, the progresses in sensing, actuation and control has largely influenced the development of legged locomotion in robotics. However, despite the latest and impressive enhancements in mechatronics, its evolution continues presenting challenges for the advancement of autonomous biped robots. Nowadays, the main requirements from a hardware point of view are listed in 2.3.3 as per [12].

- Energy-efficient and high-performance actuators with high torque-to-weight ratio and high torque-to-volume ratio.
- Highly reliable while economical sensors.
- Lightweight but mechanically strong materials for construction of the structure and mechanism.
- Fast, high computing power, and economical dedicated computer system robust against hostile situations of nature.
- Lighter power sources with greater autonomy.



(a) WAM-1 (1966)

(b) WABIAN-2R (2006)

Figure 2.6: Examples of humanoid robots.

In spite of this, it seems that the progress in legged autonomous systems will come from the hand of the introduction in the robotics field of recent areas in science as nanotechnology or smart materials. Together with this, the new findings in microelectronics and the miniaturization of computers and batteries will help overcome the current challenges.

2.4 The next generation of bipeds

It appears clear now that the evolution of bipedal locomotion has in the last decades suffered a wide-reaching change in its conditions with the entrance on the stage of robotics science and its research in walking machines. It still stands difficult, however, to foresee the paths that the advancements in artificial bipedalism will take without the risk of getting lost in speculations. Notwithstanding, the latest breakthroughs yield reasons to believe that the next generation of walking machines will resemble more and more the physiological designs developed by nature, although only to the extent that the state-of-the-art mechantronics allows. This last fact entails that a complete imitation of living biological structures is yet out of the reach of engineering, thus making its goal not to just copy, but to understand the underlying principles that drive biology and transfer them to the new synthetic creations with the existing technology.

However, the latest advances in some of the youngest areas of the engineering seem to be close to bring about a change in the so far unaltered rules of the game. The progresses in AI and in the design of genetic algorithms have arisen the possibility of not only striving for mimicking nature, but also replicating its fundamental processes, artificially accelerating

and modifying them. In the next decades, nature could stop being the only motor of evolution and from there, the possibilities are boundless.

Chapter 3

Conception and initial analysis

The process of designing and constructing a bipedal robot as the presented here is a task of great magnitude in which a big set of parameters has to be analyzed and determined aiming at the most optimal solution possible¹. The important role of the geometrical and inertial mechanical parameters in locomotion control was first proved by [70], and their relevance deserves a careful study. Thus, the sections in this chapter contain the conceptual presentation of a set of design criteria and first approaches to the construction of the RuBi prototype arisen from the study of these parameters.

3.1 Bipedal locomotion

This section contains a superficial comparative study of the characteristics of human-like walking and running patterns. The reason is to help justify in the following chapters the decisions taken during the design towards the generation of these gaits.

Although walking and running motion patterns in humans might resemble each other at first glance, they contain important variations that imply that a robot able to walk at a certain speed is not necessarily capable of running at the same speed, or running at all. The existence of the so-called "flight phase" in running but not in walking cycles is the main difference between them. This phase encompasses the period in which there is no contact between the feet and the ground and causes the main variations in energy and trajectory generation.

3.1.1 Walking and running gaits comparison

The differences in both joint kinematics and kinetics between both patterns arisen from this distinction are studied in [43], Manuscript I, for a wide range of speeds. From the cited paper it can be seen that the joint positions for a whole cycle in both walking and running patterns are very similar, specially for hip and knee. Furthermore, the joint velocity

¹Optimality here is measured in terms of the aimed goals, described in 1.2.

Index	Definition	Value	% of Stature
A	Stature (body height)	**	100
B	Crotch height	**	**
C	Femur height	**	**
D	Tibial height	Not in norm	**
E	Ankle-toe tip distance	Not in norm	**
F	Buttocks-leg length	**	**
G	Sole length	**	**

Table 3.1: Human proportions from DIN 33402-2.

profiles have almost the same shape but differ in the magnitudes, being higher the angular speeds reached for running. Similar conclusions can be extracted from a quick comparison between the torque and power profiles in a running and walking cycle. While the shapes of the functions for the three joints are the really close, with bigger differences for the ankle, the values of the curve are in average smaller for human walking than for running. As expected, the conclusion is that higher requirements for joint speed, torque and power are expected for running, specially for the knees and the ankles.

Besides, the foot strike and impact forces during the landing phase in both cases change, both in intensity and profile.

3.2 Geometrical dimensions of the frame

The selection of the final dimensions of RuBi corresponds to an iterative process in which power requirements and achieving human-like kinematics have been the main constraints when assessing the size of the robot. A first approach was taken from [43], Manuscript I, where normalized values of power required per joint and per kilogram of the structure can be found. The first iteration targeted a robot of dimensions $m = 1Kg$ and length from the hip joint to the tip toes of $L = 0.6m$ for a fully stretched leg. The final parameters from the last iteration can be found in ***

Once estimated an overall length of the structure, an initial value of the dimensions of each link were decided to be obtained based on the German section of the ISO 7250-2 [3] and the DIN 33402-2 [78] norms, following the idea of mimicking human-like motion as closely as possible. These two norms are standard references in industry and were considered a general enough source of information. Figure 3.1 depicts some of the human dimensions extracted from the norms and applied to the design of RuBi. Their names and values for male between 18-65 year old and in percentile 50 can be found in table 3.1.

3.2.1 Limbs length

The dimensions needed to create a simplified model of one leg are the limbs lengths, which are the straight line distances measured from two consecutive joints. They have

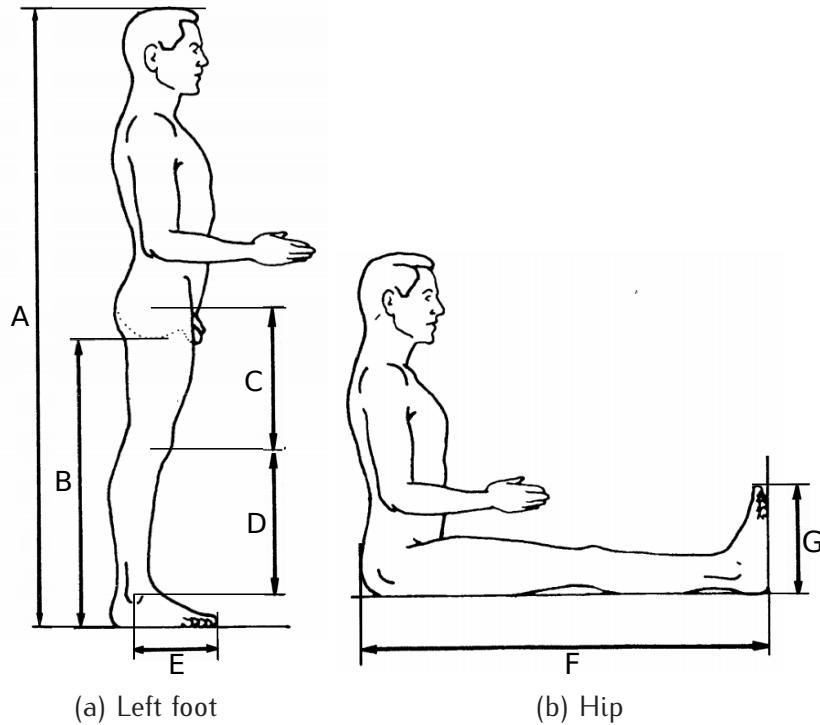


Figure 3.1: Lower body measurements used for RuBi. Picture adapted from [78].

l_i	Limb	
l_1	Hip + thigh	C
l_2	Knee + Foreleg	D
l_3	Ankle + foot	E

Table 3.2: Limbs index.

been called l_i where i is the robot link + joint as per table 3.2 and can be seen in Figure 4.3. However, the norm does not determine all of them.

Since C and E are not standard measurements in industry, they have been obtained as follows:

The Femur height The crotch height, denoted as B in the figure, has been averaged with the buttocks-leg length and the tibial height + an empirical approximation of the height of the ankle have been subtracted from the result, obtaining C .

The ankle-toe tip distance Since this distance is not standard either, it has been obtained once again adjusting the sole length with empirical measurements.

3.2.2 Hip and sole width

The hip width sitting, as defined in the norm and for the same sample group than before, has been used as starting point to define the final width of the structure of the hip. This standard measurement has been scaled down to a human of length L from hip joint to toe tip, which would correspond to $L = C + D + E$, using the proportion between stature and lower body lengths, also obtained from the norm. The same process has been conducted to calculate the implemented width of the foot sole.

No other standard measurement has been used as reference for the design, such as thigh or lower leg circumferences, since they do not affect the kinematics of the structure although they have a big influence in its dynamics. The criteria followed to model the dynamics of the robot is explained in 3.3.

3.3 Inertial dimensions of the frame

This section presents the theoretical guidelines followed during the design of RuBi regarding its dynamic model. The calculations carried out are to be found in chapters 4 and 5. The determination of the geometrical dimensions for a robot model is in general a deterministic task in which all the parameters can be selected without dependencies or limitations. However the inertial parameters of a robot will be the result of the selection of these dimensions, together with the materials utilized for the implementation and the configuration of the elements on the structure, among others. Here, the goal of the adjustment of the final inertial parameters is not to mimic the dynamics of human legs as with the kinematics, but to reduce the power requirements for locomotion while ensuring robustness and reliability. The main inertial parameters object of study here are the mass of the links, the positions of their CoM and their inertia moments. Other parameters with an influence in the model dynamics such as friction forces or delays introduced by transmissions are not discussed here due to the complexity of its determination at this stage of the project.

3.3.1 Mass of the links

The mass of each limb i will depend on the materials used, their geometry and their density together with any other component added to the link (actuators, electronics, transmissions). As explained before, keeping the overall mass of the frame as low as possible while guaranteeing the fulfillment of all the structural requirements, such as resistance and resilience, has been one of the main goals of the analyses conducted in the design of RuBi. This criteria led to the allocation of the embedded electronics off the robot and aim at small electric actuators as the lightest possible solution for the actuation. The calculations to select the limb materials and profiles are to be found in 5.2.4. The final values of mass per limb have been obtained as an approximate sum of the components that constitute them, due to the complexity of measuring them directly or using system identification techniques. They can be seen in 10.1 and they have been assumed to be concentrated in the CoM of each link for the computation of the dynamic model.

3.3.2 Mass distributions

Limbs center of mass The position of the center of mass (CoM) of each limb referred to its joint rotational axis is a function of its masses, their distributions and its geometry. Their coordinates are given referred to a local reference frame attached to each joint, with its Z axis perpendicular to the sagittal plane and its X and Y axes parallel to its equivalents in the main reference frame in the hip, shown in Figure 4.3. Its direct influence in the dynamics of the system is shown in 4.4, equation 4.9. The coordinates of the CoM of the limbs have been estimated during the design through the 3D CAD design software tool SolidWorks [69]. Their final values can be found in 10.1.

Center of mass of the frame The location of the frame center of mass plays a very important role in the stability of the robot [5]. Its theoretical placement when standing still should be in the sagittal plane of the structure, as close to the hip as possible, similar to humans (accounting that there is no torso). A robust locomotion should result in a controlled and limited motion of the CoM. A high positioning of the CoM in the structure would make it more sensitive to the actuators influence, allowing a better balance control, while a lower placement in the structure could increase its robustness against inertial phenomenas. The trade-off between these two criteria has been tried to be found. As for the limbs, the coordinates of the CoM of the structure have been computed through SolidWorks and their final values can be found in

3.3.3 Moments of inertia

The inertia tensor of each limb will be the result of their distribution of masses with respect to the joint axis. Since the kinematics analysis of the robot has been reduced to a planar case for both legs, as discussed in 4.3, the tensors have been reduced to a scalar value, their I_{zz} component for the defined reference system. The influence of the inertia moments of the limbs in the dynamics of the system can be seen in equation 4.9 where α is, as per definition, the proportion between angular acceleration around the axis and torque applied. This relationship yields another design criteria for the robot: the minimization of the inertia moments of the limbs in order to reduce the required torque applied for moving the limbs. As for the previous magnitudes, the final inertia moments have been calculated with the CAD model in SolidWorks, and can be found in 10.1. Their role in the simulation model developed in Gazebo is discussed in 6.

3.4 Joints design

Following the idea of mimicking the human lower-body structure it was decided to implement three actuated rotational joints per leg although some research was conducted on the use of passive ankles as in [6], [27] or [7]. However, the results introduced in [43] from their analysis of joint actuation for prosthetics limbs proved the importance of the actuation in the ankles for running.

The design of the joints entailed the addressing of three main areas:

- Actuator model
- Transmission system
- Implementation of dedicated compliance

3.4.1 Actuators

In humans, the actuation of the joints is mostly provided by pairs of agonist-antagonist skeletal muscles linked to the bones through the tendons [46]. The activation-inhibition of these muscles produce a lever effect on the bones that leads to its control and motion by modifying the angle between two consecutive limbs. To achieve the same kind of motion control, the implementation of a similar system through electric linear actuators was considered. However, the time constraints, their price and their complexity led to discard them and select conventional electric motors. The selected motor model will have to be able to provide a sufficient torque to generate the desired forces at the end of the link, following equation 3.1.

$$\tau = r \times F \quad (3.1)$$

This equation could be sufficient to compute the torque required by the load for a motor application of leverage. However, a study of each joint separately would not provide an accurate enough solution due to the configuration of the legs, making necessary a study of the actuation as for a kinematic chain. The analysis of the actuators requirements is to be found in 4.

3.4.2 Torque transmission

As introduced in 3.3.3, the reduction of the moments of inertia in the limbs in order to minimize the torque requirements for motion was set as a priority. This led to the study of methods to displace the CoM of the limbs as close to their joints axes as possible, thus reducing their inertias. The solution was to place the actuators, the heaviest components of the design, as close to the upper part of the frame as possible, which would also result in the allocation of the CoM of the frame upper in the sagittal plane. An schematic example of this idea can be seen in the Figures in 3.3 for the ankle actuation, where the motor has been placed under the knee axis. The same principle has been applied to the knee actuation, as can be seen in the final implementation. For the hip, however, it was not necessary to install a transmission system, but a gear mechanism whose justification is to be found in 4.

Due to the new arrangement, a powertrain was required to transmit the torque to the joints. It was decided to implement a system of 2 pulleys + belt due to its simplicity

and optimal capabilities for the project requirements. The goal of this mechanism is only the transmission of power, without any further adjustment of angular speed-torque ratios. The calculations to find the optimal dimensions of the system are contained in 5.2.1. The pitfall of this implementation, however, is the introduction of delays in the motion transmission and the natural loss of accuracy arisen from the backlash associated to the use of pulleys. Modeling these uncertainties for a classic locomotion controller based on the motion equations is proved a complex task. However it was assumed here that the ANN controllers that would drive the RuBi platform would be able to adapt to them without a model of the robot.

3.4.3 Joints compliance

The advantages of the incorporation of compliance in legged locomotion have been broadly demonstrated in [77] and [43]. Compliance in running and walking gait generation is used to reduce actuators requirements through the minimization energy consumption. Furthermore, it helps isolate the joints of the structure from the impact forces created during the landing phase, reinforcing the mechanical capabilities of the structure. This section deals with the study and selection of the configuration of the elastic actuators and their distribution in the legs. The numerical estimation of the actual springs to be used in the robot can be seen in 4.5.

Elastic actuators configuration

Several elastic actuator configurations have been defined in the literature, from which four ones have been studied here and are shown in figures 3.2. The most detailed analysis of the influence of each of these configurations in energy storage and power consumption in human locomotion has been found in [43]. In that paper, the normalized power requirements with the different configurations have been tested for a range of gaits, easing the process of finding the optimal configuration and its parameters for a specific gait and velocity. However, the fact that RuBi is not being designed for a specific gait or velocity, but as a platform to study the transition between gaits and velocity adaption, makes the estimation of the best configuration an under-determined problem. This reasoning led to the implementation of a new feature in RuBi, the full reconfigurability of the compliance in its actuators. This capability allows for the possibility of studying in the real platform the influence of the elastic actuators on its performance and its control, giving RuBi a whole new set of uses.

Compliance distribution and implementation

The results in [43] demonstrate that the implementation of compliance in any of its variants in knees and ankles, influences notably the peak power and energy requirements both for walking and running. They also seem to prove that the contribution to the reduction of energy consumption and power requirements of the hip is neglectable. These findings have

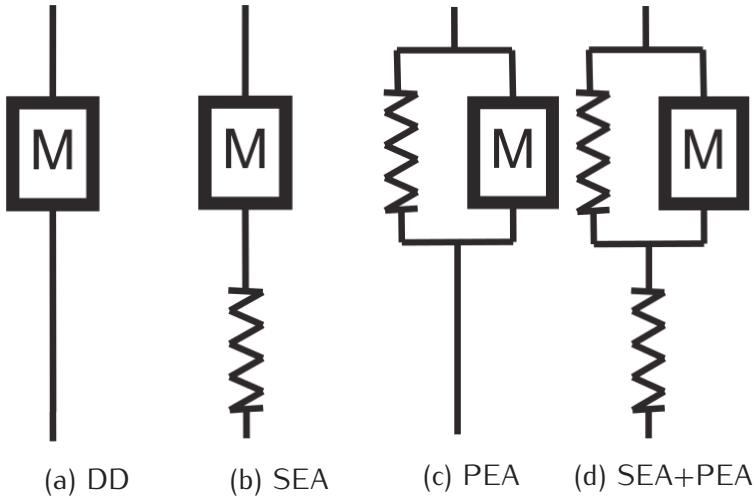


Figure 3.2: Four configurations for elastic actuators as from [43].

been used to justify the implementation of elastic actuators in knees and ankles, but not in the hip joints, actuated through direct transmission.

Several models from literature have been subject to a comparative analysis for their use in RuBi, being the main ones shown in Figure 3.3 and 3.4 for the ankle actuation (equivalent for the knees). These schematics depict the three major design choices made so far, summarized in 3.4.3. A short summary of the conclusions reached by these studies is introduced below.

- Reduction of limbs inertias by placing the actuators up on the links.
- Use of transmission to solve the transfer of power motor-joint.
- Introduction of compliance through the use of springs.

SEA 1 and 2 3.3a depicts an unidirectional SEA that has to be implemented together with a counterpart, which could be a passive spring or another actuator. In that case the actuator task would only be the retraction of the foot to reduce its angle with respect to the lower leg, while the passive spring function would imitate the calf muscle. This approach would be closer to the real functioning of the muscles in the legs, as explained in 3.1. However this configuration was discarded due to the complexity in the realization of a correct mapping between the angular displacement of the motor pulley and the ankle joint.

The configuration in Figure 3.3b was found in [14]. It solves the problem found for 3.4.3 changing the configuration of the motor and introducing a fixed pulley. The influence of the relationship between L_B and L_A in the ratio between motor and joint torques was studied for this configuration in order to find the optimal dimensions. However, this solution was discarded in favor of the current one, which demonstrated to be simpler to implement.

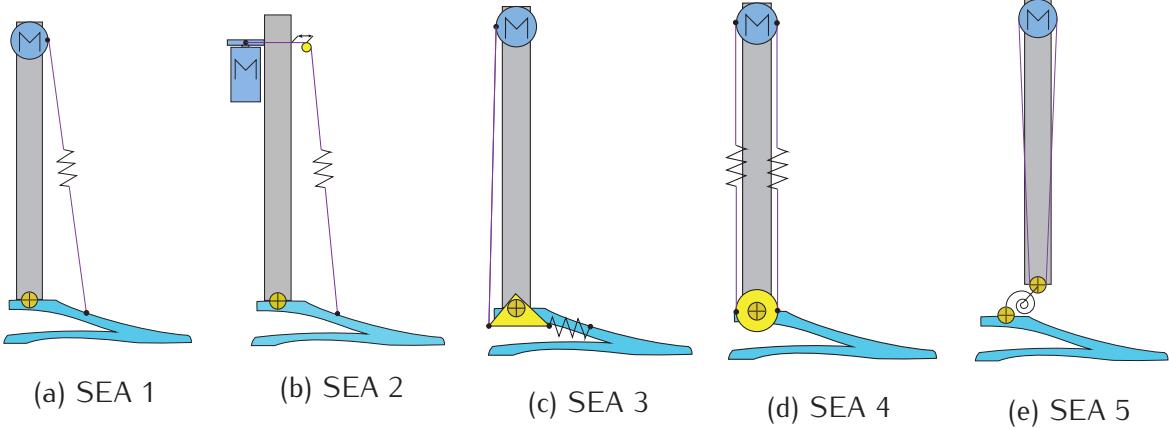


Figure 3.3: SEA configurations studied.

SEA 3 The spring configuration in 3.3c was found in [43] for a prosthetic ankle design (meaning that the configuration should be adapted for the knee). It was just used as a study case and source of ideas.

SEA 4 The configuration in 3.3d was also found in [14] and is known as bidirectional SEA. It was considered the most optimal to mimic the human musculature on the legs and its implementation through springs attached to the belts or with elastic bands (bungee cords) like the utilized in [21] was object of analysis.

SEA 5 The configuration called SEA 5 and shown in Figure 3.3e was the selected one. The original idea was taken from [27] but has been modified to ease its construction and provide more robustness to the final design. In [27] this elastic transmission is implemented with torsion bars for the joints axes, while RuBi uses torsion springs placed around more resistant axis to transmit the motion.

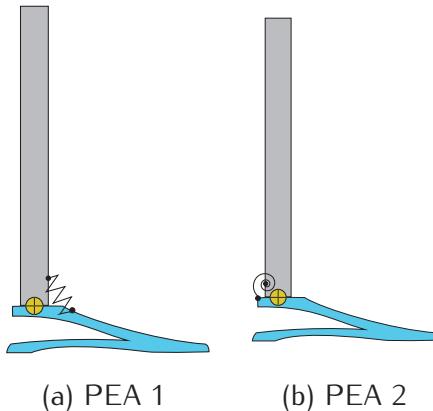


Figure 3.4: PEA configurations studied.

PEA 1 The configuration in 3.4a shows a compression spring (or extension spring depending the placement) connected within the two links. This direct connection act as a

parallel spring and has the advantage of being a very flexible system in terms of changing the physical parameters of the spring. However its design was discarded in favor of the rotational due to the previous election of the rotational series spring.

PEA 2 The configuration called PEA 2 and shown in Figure 3.4b was the selected one. This is congruent with the series spring what will facilitate its integration when designing. It also has the advantage of having common interfaces which would lead to the possibility of using the same springs for series and parallel configurations, saving thus money and time.

Chapter 4

Mathematical model of RuBi

The motivation to start the development of the mathematical model was the seek of a set of equations with which calculating the requirements for the actuators could be easily achievable. This task finally evolved towards the construction of a simple dynamic model of the robot and the formulation of the vertical jump dynamics case. The main goal was to compute the values of the necessary parameters to select joint actuators able to keep up with the requirements of the application. However, by definition, to do so the application needed to be defined to the extent that these dimensions could be obtained. Thus, this chapter contains the definition of the task RuBi is expected to be able to perform, the control model defined for it and an explanation of the assumptions made in order to simplify the resulting model. After, the necessary kinematic model of the robot is presented as an intermediary step to compute the simplified dynamic model of the robot, which is introduced next to it. All the algorithms described in this chapter have been implemented in MATLAB and the scripts can be found in the source folder attached to this report.

4.1 Actuators design

The following sections present the calculations carried out to define the load requirements of the application the actuators will be used for¹. In the present case, the operation consists in the generation of walking and running gaits for the planar robot RuBi at a wide, not-specified range of speeds. Due to the complexity of modeling such wide and uncertain task requirements, it was decided to study the case of the robot hopping on one leg assuming that the drive requirements for walking/running would be accomplished by actuators able to perform this action.

Considerations about the mechanical transformations of the motor output power before being driven to the load such as transmission mechanisms are not detailed here, but in 5.2.1. The present explanations are for the motor and gearbox selection.

Following feasibility and availability criteria, it was decided to utilize electric motors for

¹For an introduction to the actual hardware implemented in the prototype, the user is referred to section 5.1

the construction of the robot, as in many other similar cases in the literature like [33], [27] or [14]. Generally, the process of selection of electric motors entails answering the following queries about the load:

- Torques applied by the load to the motor shaft for every link τ_i .
- Accelerations involved \ddot{q}_i .
- Inertias of the masses I_i .
- Type of operation (continuous, intermittent, reversing).
- Actuator size constraints.

The first three points in 4.1 have been answered in sections 4.2, 4.3 and 4.4. The expected experiments for RuBi to be used are not meant to be of a long duration, which means that low numbers of cycles are expected per use corresponding to an intermittent application. Besides, as explained in the power requirements sections of [43] Manuscript I, high power peaks per cycle specially in knees and ankles are expected. For the above reason, and aiming at constructing as further work an on-board electronics set, batteries LiPo of 6 cells and 0.85A have been selected to supply the necessary power to the motors. Finally, the physical dimensions of the motors such as weight and volume are a key factor to be consider in the design of moving platforms. Naturally, the aim here has always been to find a motor+gearbox combination to keep the overall weight of the robot as low as possible, since it was assumed that it would be the heaviest part in the structure.

4.2 Analysis of vertical jump

Modeling the dynamics of walking gaits in a bipedal robot with 6 DOF has proved significantly complex and time consuming, even for the two-dimensional case. However, it is essential for the obtainment of the most extreme torque and rotational velocity values to apply to each joint during the motion, necessary to gauge the required actuators for the application. In order to ease the task, instead of calculating the motion equations for the walking or running cases, it was resolved to model the dynamics of the vertical jump case. This was decided under the assumption that a platform able to perform a static, vertical jump of a certain height h and determined characteristics on one leg could be able to fulfill the power requirements necessary for running, both from an actuation and a structural point of view. This assumption was built upon the findings introduced in [9] and [10].

Vertical jump dynamics

As explained in section 3.2, the dimensions of the robot and its parameters have been conceived to imitate human characteristics and capabilities. Therefore the jump analysis here will be carried out in resemblance of the human case, based on [30] and aiming at

applying the results to a robot platform, as in [16]. The jump case analyzed here can be described according to [30] as a static, squat jump over one foot with take-off and without counter-movement (SJ-NAS). Three assumptions have been made regarding the bodies in contact during the landing phase of the robot as listed in 4.2.

- No slipping occurs.
- Absence rebounding.
- There are no structural deformations.

The whole jump cycle has been divided into two phases with different analysis: an impulse and an aerial period.

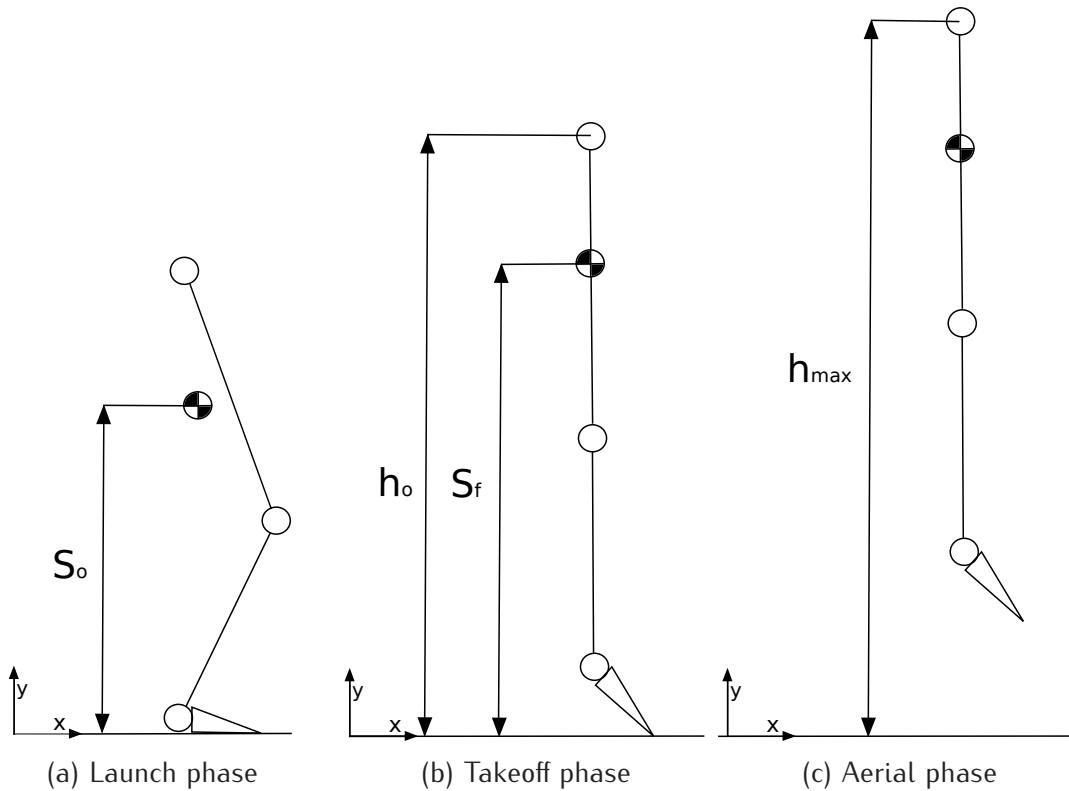


Figure 4.1: Jump phases without fall and landing.

The aerial phase The period from the take-off of the toes to the instant in which the maximum height h_{max} has been reached, which takes place between the instants represented in 4.1b and 4.1c. The rest of the cycle, including fall and landing are not described here. During the whole period the robot is considered a rigid solid. This phase is analyzed following the equations of a static projectile motion for the vertical launch case, expressed in 4.1 and 4.2.

$$h(t) = V_o t \sin(\theta) - \frac{1}{2} g t^2 \quad (4.1)$$

$$mg\Delta h = \frac{1}{2}m\Delta V^2 \quad (4.2)$$

The equation 4.2 is an energy balance analysis for the flight phase, that can be rearranged as showed in 4.3

$$\Delta V = \sqrt{2g\Delta h} \quad (4.3)$$

This equation can be used to input the desired displacement in the CoG of the robot during the free flight period and obtain the necessary velocity at take-off V_{toff} , which is the one the robot has to reach in the impulse phase. This can be done since $V_{final} = 0$ and therefore $\Delta V = V_{toff}$.

The impulse phase From the initial position at the beginning of the jump, depicted in 4.1a, to the instant when the toes of the standing foot take off the ground and the ground reaction force ceases, shown in 4.1b. The change in momentum in the robot, of total mass m , caused by the application of a vertical force with the foot to the ground F during a period t is given by equation 4.4, derived from the second law of motion of Newton.

$$Ft = m\Delta V \quad (4.4)$$

Substituting equation 4.3 in 4.4, the necessary impulse to reach the desired maximum height can be computed. However, since the force applied to the mass and the application period cannot be decoupled from the result of the equation, an energy analysis has to be carried out together with a posterior optimization of the two values for the desired design of the robot. Solving equation 4.4 for a small range of application time values and after computing 4.3 with increasing values of Δh in the range [0.01, 0.1] meters yields a reciprocal graph whose positive quadrant can be seen in Figure 4.2. However, the obtained graphs do not propagate to an infinite value of application time, existing a boundary farther than which, the impulse will not produce enough variation in the momentum to launch the body.

$$W = \int_{t_{toff}}^{t_f} F(t) dt = \frac{1}{2}m\Delta V^2 = F_{min}S \quad (4.5)$$

Equation 4.5 describes how the work performed by the robot during the impulse phase can relate to the change in its kinetics energy. This equation, derived also from Newton's second law would only apply to rigid bodies without internal degrees of freedom, which is not the case. However, it has been assumed that the displacement given by S is the vertical translation of the CoM of the robot during the impulse and that it is small enough to be a valid assumption in order to use this procedure. From Figure 4.1, the displacement of the CoM during the impulse is given as $S = S_f - S_o$. Knowing the value of S , the minimum necessary force can be obtained, and hence its corresponding application time

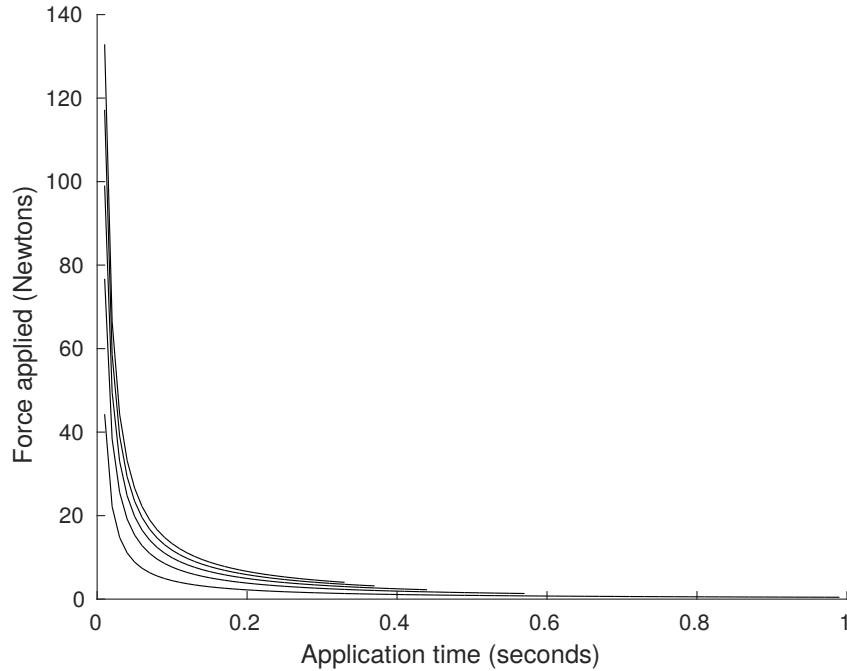


Figure 4.2: Force-application time values for the same work magnitude.

on Figure 4.2 for a given Δh . The displacement of the CoM, S will depend on the initial configuration of the robot prior starting the impulse, and its final one. Furthermore, for a given Δh , a range of possible values of the ground reaction Force F can be computed within the limits established and a time can be obtained. Thus, for a determined Δh and t values, a F has to be applied by the foot again the ground. In order to translate that force to joint effort, the dynamic model of the robot is required.

4.3 Kinematic model

The kinematics model of the robot describes the motion of every joint in time disregarding dynamic parameters. It is used here to compute position, velocity and acceleration of the limbs for given trajectories of the toes, which are the contact point with the ground in the presented study case, through their forward kinematics. Also the inverse kinematic model is constructed as a tool for the posterior computation of the dynamics model and its application to model the necessary actuators. The model has been obtained through direct application of trigonometry instead of using the D-H parameters given its simplicity. To do so, as explained before the case has been reduced to a two-dimensional study of one leg, where each robot leg has been analyzed as a kinematic chain of three degrees of freedom. The main reference frame has been placed attached to the hip with its Y axis parallel to the ground, as if this was the base of a robotic arm, and the toe was the tool, has shown in Figure 4.3. This representation has been chosen to ease the construction of the mathematical model. The Figure 4.3 depicts the schematic representation of only one leg, since the other one is equivalent. The system is said to be in single support phase since there would be just one foot in contact to the ground. In the analyzed case,

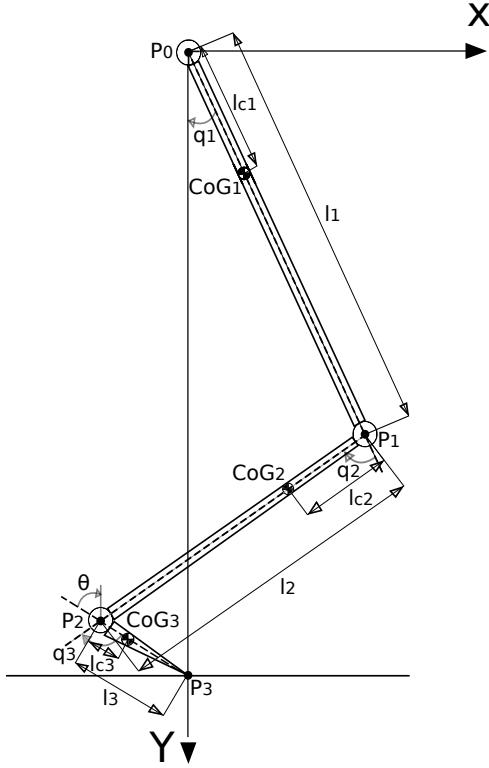


Figure 4.3: Coordinate definitions for the schematic model of one leg.

the swinging leg (not represented in Figure 4.3) would be static and could be considered an additional link with no degrees of freedom. The variables q_i represent the angles of the joints with respect to their fully-stretched positions to simplify the obtainment of the forward kinematic model, while the angle Θ is used to express the orientation of the ankle in an absolute manner. From Figure 4.3, the forward kinematic model in 4.6 can be easily obtained.

$$\begin{aligned}
 x_0 &= 0 \\
 y_0 &= 0 \\
 x_1 &= l_1 \sin(q_1) \\
 y_1 &= l_1 \cos(q_1) \\
 x_2 &= l_1 \sin(q_1) + l_2 \sin(q_1 + q_2) \\
 y_2 &= l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) \\
 x_3 &= l_1 \sin(q_1) + l_2 \sin(q_1 + q_2) + l_3 \sin(q_1 + q_2 + q_3) \\
 y_3 &= l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) + l_3 \cos(q_1 + q_2 + q_3)
 \end{aligned} \tag{4.6}$$

The variables x_i, y_i represent the position coordinates of the joints in the right leg, named P_i in the schematic in Figure 4.3 and P'_i for the left leg (not depicted). Once computed, the linear velocity and acceleration of every joint can be obtained by derivation of the forward kinematic model and represented as \dot{P}_i and \ddot{P}_i .

The model presented is the most basic one and does not contain the degrees of freedom that could be introduced by springs in the actuators. This is due to two reasons:

- Their configuration can be changed to series, parallel or none in both knees and ankles.

- The study of the jump case regarding the parameterization of the actuators has been carried out in the most disadvantageous scenario from the energetic point of view, which in this case means no springs.

Following the same approach, the inverse kinematic model can be computed for the right leg, yielding the equations 4.7.

$$\begin{aligned} x_2 &= x_3 - l_3 \sin(\Theta) \\ y_2 &= y_3 - l_3 \cos(\Theta) \\ q_1 &= \arctan\left(\frac{x_2}{y_2}\right) - \arccos\left(\frac{x_2^2 + y_2^2 + l_1^2 - l_2^2}{2l_1\sqrt{x_2^2 + y_2^2}}\right) \\ q_2 &= \pi - \arccos\left(\frac{l_1^2 + l_2^2 - x_2^2 - y_2^2}{2l_1l_2}\right) \\ q_3 &= \Theta - q_1 - q_2 \end{aligned} \quad (4.7)$$

And equivalently to the forward kinematics, the inverse kinematic model can be derived to obtain the velocities and accelerations in the joint space \dot{q}_i and \ddot{q}_i .

4.4 Dynamic model

The goal of this step is to calculate the relationship between an external force applied to the toe (ground reaction force while jumping) and the necessary torques in the joints for dealing with that external disruption. This relation can be expressed as a set of second order differential equations represented for the general case as in equation 4.8.

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau \quad (4.8)$$

Where $B(q)$ is the inertia matrix, $C(q, \dot{q})$ contains the centrifugal and coriolis acceleration terms and $g(q)$ represents gravity, as in [2] and [11]. It must be remarked here that the constructed model does not introduce joint friction terms. For an open kinematic chain as this, three methodologies to obtain the above equation where object of study:

- A simplified Euler-Lagrange algorithm, as introduced in [19], which makes use of the Lagrangian formulation to describe the behavior of the system through work and energy.
- The so called Energy Method, presented in [36] and consisting in finding the relation between the force in the end-effector and the joint torques through the Jacobian of the kinematic chain.
- The Newton-Euler algorithm, through which the dynamics of the system can be expressed in terms of forces and moments applied in each member of the chain.

It was finally decided to apply the third option due to the fact that it was faster to implement than the E-L algorithm and more reliable than the Energy method. The Newton-Euler

algorithm is founded in classical mechanics and its a recursive method that computes in two steps the velocities and accelerations of every component of a kinematic chain and their forces and torques on the joints. In 4.9, the equations of motion for an individual link are shown.

$$\begin{aligned} f_{i-1,i} - f_{i,i+1} + m_i g - m_i \dot{V}_{ci} &= 0 \\ N_{i-1,i} - N_{i,i+1} - (r_{i-1,i} + r_{i,Ci}) \times f_{i-1,i} + (-r_{i,Ci}) \times (-f_{i,i+1}) - I_i \dot{\omega}_i - \omega_i \times (I_i \omega_i) &= 0 \\ i = 1, \dots, N \end{aligned} \quad (4.9)$$

These equations contain the coupling forces and moments applied to the link by the immediate ones, however, they cannot be used with them. ω_i and $\dot{\omega}_i$ represent respectively the angular velocity and acceleration vectors of link i and can be obtained from the joint velocities and accelerations, computed in the previous section, as in equation 4.10. I_i are the inertia matrices of the links, calculated in SolidWorks for the final design of RuBi.

$$\begin{aligned} \omega_i &= \omega_{i-1} + \xi_i \dot{q}_i Z_{base}^i \\ \dot{\omega}_i &= \dot{\omega}_{i-1} + \xi_i [\ddot{q}_i Z_{base}^i + \dot{q}_i \omega_{i-1} \times Z_{base}^i] \end{aligned} \quad (4.10)$$

The closed-form equations must be derived from them so that they can be applied to the algorithm. This is done by substituting the coupling forces in the final set of 6 equations obtained for $N = 3$ and substituting 4.11 in the resulting equations.

$$N_{i-1,i} = \tau_i \quad (4.11)$$

Equation 4.11 is valid only for the planar case. After these steps, a set of three equations of the form 4.8 is obtained. As parameters, they contain the masses of the links, their vectorial positions, the gravity term and their inertia matrices.

This set of equations calculates the necessary torques on the joints as a function of the joints displacements and the external forces and torques applied on the system, as represented in 4.12.

$$\tau(t)_i = f(q_i(t), \dot{q}_i(t), \ddot{q}_i(t), \tau_{ext}, F_{ext}) \quad (4.12)$$

However, in order to use it the functions that model the trajectories of joints in the joint space must be approximated. For the ideal static, vertical jump case under study here, the movement of the toe has been constrained to a vertical displacement along the Y axis in order to simplify this task. Thus, the trajectory of the end-effector of the kinematic chain, P_3 in Figure 4.2, can be easily approximated as a function of the form shown in

$$\begin{aligned} x_3(t) &= 0 \\ y_3(t) &= y_3(t_o) + \frac{y_3(t_f) - y_3(t_o)}{(t_f - t_o)(t - t_o)} \\ \theta(t) &= \theta(t_o) + \frac{\theta(t_f) - \theta(t_o)}{(t_f - t_o)(t - t_o)} \end{aligned} \quad (4.13)$$

Therefore, if equation 4.13 is used as the input to the inverse kinematic model, the joint displacements will become dependent on the linear displacement of P_3 , and equations 4.12 will be rewritten as 4.14.

$$\tau(t)_i = f(P_3(t), \tau_{ext}, F_{ext}) \quad (4.14)$$

4.5 Springs influence in the actuation

This section deals with the calculations carried out in order to model the influence of the implemented compliance in terms of energy efficiency and joint torque requirements. As explained in 3.4, the selection of the elastic actuators installed and their configuration on RuBi has been left as a customizable feature for research purposes. This, together with the fact that RuBi has not been designed to optimize a specific type of gait made pointless the calculation of optimal values of springs. Thus, the following is just a theoretical frame presented to provide an easy method to compute the influence of the springs in the performance once the user has selected them. It was meant to be used for the experiments devised on compliance influence in hopping motion, presented in 8.

4.5.1 Torque and energy contribution of passive actuators

All the springs that can be implemented in RuBi are torsional, as shown in 5.6, whose general equations for torque and energy storage are shown in 4.15.

$$\begin{aligned}\tau_s &= -K\Delta\theta \\ U_s &= \frac{1}{2}K\Delta\theta^2\end{aligned}\tag{4.15}$$

The contribution of the springs torques to the system actuation is therefore modeled introducing the torque equation in 4.15 on the right side of 4.8, which would yield equation 4.16. Where the torques vector has been divided into the motors and springs inputs.

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau_M + \tau_s\tag{4.16}$$

The values of $\Delta\theta_i$ can be obtained through forward kinematics for given trajectories, while τ_i can be computed from equation 4.8 given the ground contact force model, which is the only external force applied to the system. Thus, the first formula in 4.15 can be used to calculate the optimal K for a desired joint trajectory and force model, for instance. Knowing k_i and $\Delta\theta_i$, the energy stored per gait cycle by each spring can be computed and utilized for studying their influence on the performance of the robot.

4.5.2 Joint kinetics

The motor power requirements for each joint can be calculated through equation 4.17 for direct drive transmission. This formula does not include inertias, frictions or any other efficiency coefficients.

$$P_m = \dot{\theta}_m \tau_m\tag{4.17}$$

By definition of 4.17, the value of P_m can be positive (motor thrusting) or negative (motor dumping). Thus, the peak power in the motion cycle can be computed as its maximum absolute value. Besides, the equation of motor power is used in 4.18 for calculating the overall energy requirements for a whole motion cycle, employing only absolute values as well.

$$E_{cycle} = \int P_{m+}(t)dt + \int P_{m-}(t)dt \quad (4.18)$$

Influence of elastic actuation

The above presented are general formulations. However, the introduction of elastic transmission between motor and load through springs modify the computation of motor power. The changes resulting from implementing the configurations shown in Figure 3.2b, 3.2c and 3.2d yield equations 4.19, 4.20 and 4.21 respectively, as per [43]. For a complete and detailed description of the parameters in these equations, the reader is referred to the cited paper.

$$P_m = \tau_m \left(\dot{\theta}_t + \frac{\dot{M}_{ex}}{K_s} \right) \quad (4.19)$$

$$P_m = (F_{ex} + (K_p \Delta \theta_t)) \dot{\theta}_t \quad (4.20)$$

$$P_m = \left(F_{ex} + (K_p \Delta \theta_m) \left(\theta_t + \frac{\dot{F}_{ex}}{K_s} \right) \right) \quad (4.21)$$

4.6 Model-based controllers

As introduced in the previous sections, the equations of motion derived for RuBi can be used to compute the necessary output values of its actuators in order to perform an input toes trajectory and external forces, as expressed in 4.14. By definition, this could be sufficient to, appropriately used, be utilized as the base of a controller for the robot, without accounting the own dynamics of the actuators. As an example of the above said, and despite the early stage in its development in which this model is left, some data has been computed and plotted for proving the concept. The only trajectory planner current implemented computes vertical straight paths for the toes. However, more complex trajectory equations could be used to achieve different locomotion patterns.

An initial and final toes position has been inputed to 4.13 to define an example trajectory for the toes in one leg, together with a random value for Δh . The output of the kinematics computed for $i = 1, \dots, N$ steps is shown in Figure 4.4, where it has been rearranged for

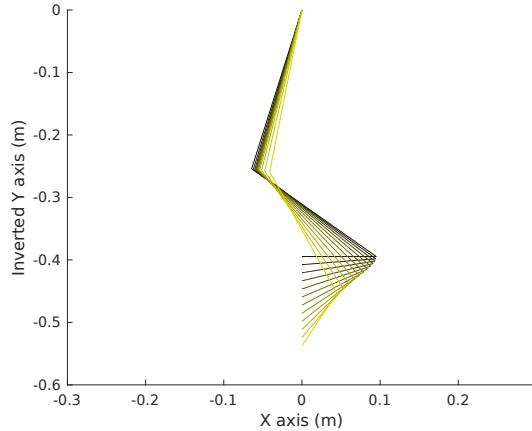


Figure 4.4: Joints trajectories from kinematics model.

visualization. It can be seen that the toes trajectory does not cover the full range of a vertical jump.

$$\dot{\theta}_{i,j} = \frac{|\theta_j(t_i) - \theta_j(t_{i-1})|}{t_{step}} \quad (4.22)$$

For the trajectory described and Δh , the impulse equation in 4.4 has been computed and the values (F_{min}, t_{max}) from 4.5 have been used as input to the dynamic model. The result of computing 4.8 for the same time steps is shown in 4.5a, together with the solution for equation 4.22 in 4.5b, calculated in an analogue way.

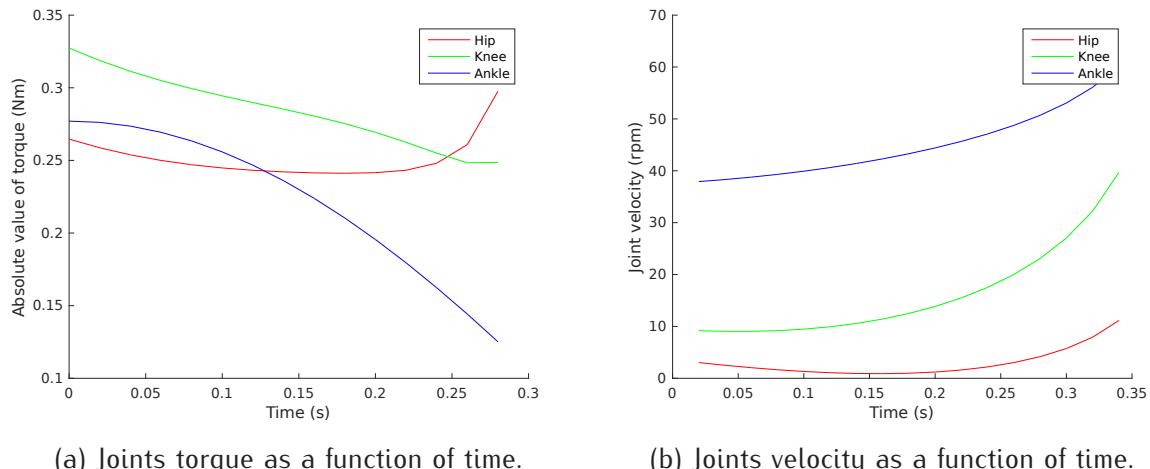


Figure 4.5: Joint torque and velocity as a function of time.

The ranges in which the results in 4.5a and 4.5b are found seem consistent with the obtained during the experiments detailed in 5.1.2. However, the analysis of their accuracy would entail a more precise construction of the mathematical model and study of kinematic chains dynamics which laid out of the scope of this project since it was not considered a priority goal.

Chapter 5

Mechatronic design

The results of the conceptual studies conducted in chapter 3 laid the goals and criteria to lead the design process of the first prototype of RuBi. The actual implementation of these ideas in the fields of electronics, mechanics and software is described here. When designing from a holistic point of view, these three areas combined give a positive synergy which is the burden of this chapter. The first of the three sections starts with the design of the electronic hardware: from the selection of the motors based on 3.4 and 4 to the expansion interfaces installed in order to reduce the weight and finally the sensory feedback system. It follows the mechanical design of the limbs components, in which the constraints defined in 3.2 and 3.3 are applied, together with the structural and production-oriented calculations required. It concludes with a presentation of the CAD design process carried out, in which a comprise between the components ideal capabilities and the current production capacity in this project has been tried to be reached. In the last section, devoted to the software-related development except for the simulation environment, the architecture implemented for the robot control system is presented.

5.1 Electronics

5.1.1 The electric actuators

In chapter 4, the necessary characteristics of the actuators have been calculated. In this section, the resulting theoretical requirements are used to select the final motor + gearbox combination utilized. All the documentation regarding the control of the actuators software-wise is to be found in section 5.3.

Flat BLDC Maxon motors

It must be mentioned at this point that the actuators and their interface were assumed at the beginning of the project to be a very hard constraint in the design from an economic point of view. This means that the conception of the robot structure has been influenced

by this criteria towards the adaption of the final prototype characteristics (such as final size or mass) to the application range of the available motors at our disposal. This fact has converted the design in an iterative process of optimization whose final result is a robot that matches the available actuators and not the other way around, as it should be in theory. In the view of this, the brushless DC motor + gearbox present in the Locokit robot construction kit, introduced in [41] are used in the RuBi prototype.

The flat motors model is 339260 from Maxon motor, whose datasheet can be found in [47], and the planetary gearhead is the number 143976 in datasheet [48]. The electromechanical constants of the motors, together with its nominal supply values or the output power and torque of both the motor and the gearbox can be found in these documents. However, the electronics of the motors are designed to constantly overdrive them at 24V, which has been taken into account when calculating their output. Furthermore, each motor counts three hall effect sensors able to provide accurate relative position measurements.

BLDC motor boards

Each BLDC motor in the Locokit comes with a motor board able to control it, designed for 24V and 48W. They consist of a 48MHz ARM7 processor for time critical control and motor commutation, as stated in [42], together with 4 general purpose I/O inputs for local sensor interface. Furthermore, they have two available 8-pin interfaces for the motors, one of them with a standard flex connector used in most of Maxon flat motors.

Extension PCBs

Following the idea of reducing weight and inertias in the structure as explained in chapter 3, it was decided to place all the electronics off-board. In order to extend the existing motor flex interfaces, a simple extension PCB was manufactured for each device. The boards have been designed with Eagle following the requirements of current when sizing the width of the paths given by the supplier. The design lacks of vias which reduces the complexity and facilitates the manufacturing. The board for the left leg, whose schematic can be seen in 5.1¹, contains a flex connector, like the one originally found on the motor boards, mapped to an 8-pin Molex connector where the wiring to the BLDC board is connected.

5.1.2 Suitability of the motor model for the application

The algorithm designed to prove if the selected motor model fulfills the requirements of the application has been called Algorithm 1 and it is detailed in 5.1.2. The theoretical framework constructed in 4 has been applied here to prove if the motors described above (without accounting on springs or indirect transmission) can be used to perform the vertical jumps described in 4.2. And if so, which height can be reached.

Algorithm 1:

¹For the right leg the schematic has been mirrored

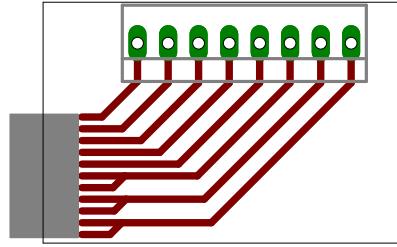


Figure 5.1: Left leg extension PCB schematic.

1. Manually set the values of $P_3(t_0)$ and $P_3(t_f)$ to use in equation 4.13.
2. Compute the inverse kinematics model for the given trajectory equation. This yields as outputs $q_j(t)$, where $j = 1, \dots, N$ being $N = \text{number of joints}$.
3. Compute forward kinematics to obtain $P_j(t)$.
4. Derive the two mentioned models and apply equation 4.10 to obtain \dot{P}_j , \ddot{P}_j , ω_j , ω_j , $\dot{\omega}_j$,
5. Set Δh and use the impulse equations 4.3 and 4.4 to obtain a set of couple values of (F_i, t_i) as in Figure 4.2.
6. For each couple until (F_{min}, t_{max}) , obtained from 4.5, compute the torques $\tau_{i,j}$ for each joint through 4.8 and $\theta_{i,j}$ as in 5.1.²
7. Compute the Torque/Speed curve for the motor + gearbox model as per equation 5.2.
8. Plot the obtained pairs of values $(\dot{\theta}_{i,j}, \tau_{i,j})$ over the motor curve and analyze the results.

$$\dot{\theta}_{i,j} = \frac{|\theta_j(t_f) - \theta_j(t_o)|}{t_i} \quad (5.1)$$

$$\tau_m = \tau_{stall} - \omega_m \left(\frac{\tau_{stall}}{\omega_n} \right) \quad (5.2)$$

Use case As with any kind of DC motor, the goal is that the operation points lay under the torque/speed curve for a given application. In this case, the operation point to study has been chosen to be the initial instant of the launch phase during the jump, given by $t_0 = 0s$, because it has been assumed to be the most requiring one of the whole jump cycle. The input data for the test conducted with the algorithm can be seen in 5.1.2.

²Eq. 5.1 introduced the assumption that the joint velocities are constant for simplicity

$$P_3(t_0) = \begin{bmatrix} 0 \\ 0.3508 \\ -\frac{\pi}{2} \end{bmatrix} \quad P_3(t_f) = \begin{bmatrix} 0 \\ L \\ 0 \end{bmatrix} \quad \Delta h = 0.05$$

The results can be seen in Figure 5.2 for a jumping leg. It can be seen that, for the obtained $(\theta_{i,j}, \tau_{i,j})$ values for the given Δh , the closer they get to the theoretical (F_{min}, t_{max}) , the more the application points approach the lower-left corner of the graph. The aimed situation here is that in which the application points for the three motors lay under the motor curve, which occurs for values next to the couple $(4.1191N, 0.1700s)$

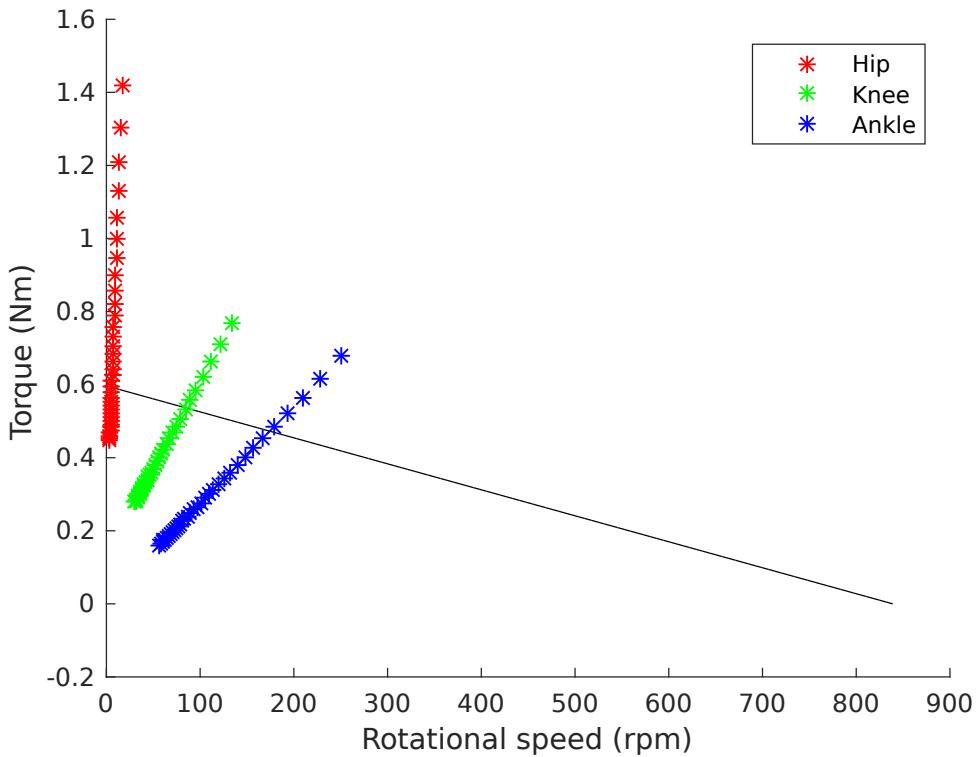


Figure 5.2: Results of Algorithm 1 for the given inputs (not all the $(\theta_{i,j}, \tau_{i,j})$ pairs are plotted).

The transmission on the hips The results from the presented algorithm helped notice that the presented motor model would not be able to accomplish the requirements of the application on the hip joints, due to its, in general, lower velocity and higher torque values. Thus, the algorithm was used to approximate the required ratio of the gear system described in 5.2.2, used to adequate its output to the task. The final ratio implemented and used for the results in 5.2 is $w = 2$. However, it was calculated for the geometrical and inertial parameters of a different iteration than the last one, resulting in a non-optimal value for the final robot. A repetition of the process yielded an optimal gear ratio of $w = 1.2$, for which the best application points of the three motors would be closer to the motor line.

The presented method does not aim at providing exact results since its based in several

Controlled joint	Register address
Right hip	7
Right knee	11
Right ankle	19
Left hip	5
Left knee	8
Left ankle	4

Table 5.1: Internal register addresses of motor boards.

assumptions and simplifications from its basis. It was conceived due to the necessity of assessing the utility of the existing actuators to the designed application. Ideally, its results would have been tested by comparing them to the real data collected from the experimentation with the final prototype. However, the delay in some essential components of the robot prevented from testing and improving the model used for the algorithm, together with its validity.

5.1.3 Embedded electronics

The Locokit main processor is a standard Gumstix Overo Air board with a 600MHz OMAP3, 512MB RAM and WiFi [52]. It is mounted on a expansion board that provides interfaces as I2C, USB or GPIOs along with some on-board sensors not used here and the connection to the power board. The power board was designed to allow a stable voltage of 24V and a maximum current of 10A with an efficiency of 90–95%, as stated in [42]. It also makes available the connectors for the six-wire bus connection to the motor boards. The communication between the main processor and each ARM7 in the motor boards is carried out through a system of common registers that are updated in every cycle of transfer, and for which each BLDC board only reads its assigned slots. The addresses of these registers are given through the external switches on the bottom of every motor board. The current configuration is given in table 5.1.

Each ARM7 reads the motor commands and writes the sensory feedback information on its register, reducing computation load in the main processor and simplifying the extension of the system with more motor boards to a simple connection. As a result, the LocoKit electronics platform provides a reliable power supply and time critical control of actuators, an easily extensible and reconfigurable architecture, communication capabilities with external equipment and possibilities for sensory feedback handling. All these features made it optimal for the expected requirements of the RuBi robot hardware, and led to its use on the project.

5.1.4 Sensory feedback

As stated in the initial description of the project in 1.1, the first prototype of RuBi has been designed to provide the necessary capabilities to be controlled by an existing neural

controller developed in [6] and already tested in the DACbot robot. The cited control algorithm requires as inputs the angular positions of all the joints it can actuate, besides ground contact signals from the feet for its reflex-based controller part. All the documentation regarding the handling of the sensor readings software-wise is to be found in section 5.3.

Joint position information

To provide the physical readings of the angular positions of the joints, the built-in hall sensors in the motors are utilized. Three wires transmit the hall effect sensors signals to the motor board for each joint, where they are written in the internal register and transferred to the main processor for its posterior treatment.

Ground contact signal

One contact switch model Omron D2F-01F-T has been placed on the edge of the sole of each foot, under the heel in order to detect when the feet are standing on the ground. Their wiring has been extended to the main processor board, but the necessary pull-up resistors have not been implemented since the input pins on the processor could not be set up. The mapping between the processor's GPIOS handlers and the physical pins on the board could not be found. Therefore this last step is left as further work. Alternatives to the use of the main board pins are discussed in chapter 11, in case they cannot be used.

5.2 Mechanics

5.2.1 Pulleys and belts

Section 3.4 has been dedicated to analyze and define the kind of motor configuration and transmission system which is going to be used for each joint. In the case of the knee and the ankle, the combination *motor + gearbox + belt and pulleys* was selected, as explained before, following inertial-reduction criteria. The transmission system designed to convey the motion from the motor shaft to the joint consists finally in the *pulley+belt* presented here plus a torsional spring in series used to transfer the rotation movement from the joint pulley to the limb, as discussed in 3.4.3. As a result, an interface between the pulley and a spring holder was needed. Hence, the design of the pulley itself has been studied in terms of two factors: *precision and backlash reduction* and *integration with the series torsional spring*.

System backlash reduction

All the pairs of pulleys are meant to have the same diameters since they are not planned to be used for torque-speed ratio modifications between actuators and joints. Thus, the final value of this dimension obeys only to other components dimensions constraints. With that degree of freedom, the aim of the design process is to optimize the pulley in order to minimize their associated backlash, defined as the clearance between timing belt teeth and timing belt pulley grooves. Despite the fact that the platform is going to be used mainly with adaptive controllers (e.g. ANN-based) and therefore the mechanical optimization is not a priority, the reduction of mechanical uncertainties is always sought.

After analyzing the current solutions in the market, several non-backlash solutions were found. The optimal one seemed to be the Gates GT3 Synchronous Belts³, able to fulfill the requirements of the presented application. The withdrawals of this option were the time constraints for ordering such parts and the increase in the final price of the product. But also, the integration with the series torsional spring.

Integration of the pulleys with the series torsional springs

An alternative solution was to design and manufacture the pulley, which would allow a complete control of the design and manufacturing process, thus yielding the possibility of integrating in a single part the pulley and the spring holder. At first, the GT3 design from Gates was intended to be utilized as a model. However its design, which is described in U.S. Patent Number 4,515,577, is patented and not open to the public. As an alternative, the ISO 13050:2014 [68], following the type T, was used to create the model of the customized pulley. This choice was based on its focus in efficiency and reduction of backlash, together with its optimality for accurate movements with high torques and low speeds, as the intended application here. The physical properties of the pulleys as the number of teeth, width, etc... were selected according to the ISO 5295:1987 [24], and as mentioned before, with the view on their manufacturability.

The interface between the described pulley and the limb was created as a torsional spring leg holder attached to the pulley's side. An equivalent piece was placed on the joint, turned 90° for holding the other leg of the spring, which is meant to be coiled around the axis of the joint. Based on the two ISO norms cited before and after some iterations based on experimental tests, the pulley T2,5 of 19 teeth gave the expected behavior. In Figure 5.9e a detail of the final pulleys can be seen. Figure 5.9d shows the pulley side of the series spring interface.

Belts selection

From 5.2.1 a system pulleys-belt using T2,5 teeth profile was selected. For the implementation of the belt system, an open belt whose initial tension can be adjusted with a zip-tie is chosen. This allows to adjust the tension at any moment without the need of

³<http://www.gates.com/products/industrial/industrial-belts/synchronous-belts/powergrip-gt3-belts>

further tools. Besides, it facilitates the system maintenance. However, no studies were conducted on the right amount of tension to apply to the belts. The complex dependency of the tension values on the application requirements led to their empirical adjustment as the most suitable option.

5.2.2 Gears

When sizing the motors in 4, the same kind of motor for all the joints was chosen reducing then the number of unique parts and increasing the modularity. However, a reduction of 2 : 1 was found to be needed for the hip joints during the calculations carried out in 5.1.2. As defined in the analysis chapter 3, the hip will be designed without any passive actuator and the motor will be as close as possible to the joint, which made a geared transmission the most appropriated option.

The gears have been optimized based on a trade-off between manufacturability and performance. A small teeth size was sought in order to reduce the backlash and assuring that there are always teeth in contact. The size of the teeth has been defined by the smaller precision of the 3D printer in which they were printed. The gears have been designed with the *Coarse Pitch Involute 20 deg* standard due to its easy manufacturability. Despite a *herringbone* gear was considered, the simple *spur* was chosen since it facilitated the assembly and could be adapted to unpredicted problems in the manufacturing and assembly processes. In the Figure 5.3, a detail of the teeth is shown.

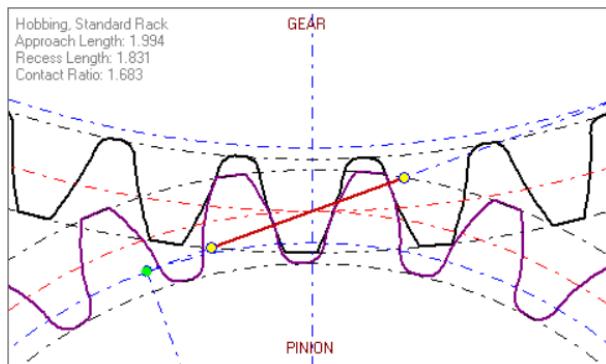


Figure 5.3: Teeth detail from pinion and gear.

There was not a FEM or stress analysis and the experimental method was preferred due to the current unpredictable mechanical efforts of the 3D printed parts. In the appendix .5, all the properties of the gears are to be found.

5.2.3 Landing impact force

The assessment of the desired mechanical characteristics of some of the components in the limbs requires the calculation of the internal forces they will be subjected to during the motion. This process would normally entail the creation of an impact model for the feet and the ground, which due to its complexity has been left as further work. To approximate

Parameter	Value
Total mass [kg]	1.5
Jumping height [m]	0.1
Impact displacement [m]	0.005

Table 5.2: Input parameters for calculating the impact force.

the output of this model, the calculations of the energy transferred to a landing leg in collision with the ground have been carried out using the model of falling object hitting the ground. The assumptions made for the scenario are listed in 5.2.3.

1. No bouncing or slipping
2. No losses of energy between states
3. No compliance on the leg
4. No deformation on the ground surface
5. All the initial potential energy before the fall is dissipated in the impact

This energy is then transmitted from the first contact point, the footprint, to the rest of the system, causing stresses that must be absorbed. The components of the system must receive that energy under a controlled behavior – elastic deformations- ensuring a longer service life of the materials. Thus, some input parameters to calculate the impact force are assumed. From this force all the consecutive components in the deformation chain will be sized. Despite the deformation is of the whole system, the security coefficient assumed here is going to be the calculation of all the components for that maximum force.

From the formula of the mechanical energy:

$$E_{mechanical} = mg\Delta h + \frac{1}{2}mv^2 \quad (5.3)$$

For a free falling object in the defined scenario, it is assumed that all the potential energy on the first instant is transferred to the crash without intermediary loses. This energy is then translated into force by supposing a deformation of the whole body as expressed in the equation 5.4.

$$F_{impact} = \frac{mg\Delta h}{d_{impact_displacement}} = 294.40N \quad (5.4)$$

The equation 5.4 gives the force for sizing all the components. Based on the input parameters defined in the appendix .1 which are shown in the table 5.2.

5.2.4 Limbs profile

Based on the requirements of weight and its desired distribution defined in the structural analysis of the robot, in the joints study in 3.4 it was decided to place the actuators as further up in the limbs as possible. This left the functionality of the limb as the frame to hold the joint components on both ends plus the actuator, also working as a placement for the wiring. Thus, a lightweight section that satisfies the conditions of deformation and stress was needed. Carbon fiber seemed to be an ideal material to achieve these conditions of weight and stress, so a generic quantitative analysis was carried out to calculate the optimal profile with this material between the ones offered by the available provider. The provider was chosen due to previous experiences that the Mærsk Mc-Kinney Møller Institute had with carbon fiber orders.

The section profile offered⁴ are: *Rod*, *Tube*, *Box*. The *Stripe* and the *Angle* were discarded due to their asymmetrical geometry that would lead to less predictable scenarios. The study case is shown in the Figure 5.4, where the impact force could be decomposed into pure bending and pure compression effort components. Since the resistance of the carbon fiber in pure compression is greater than in under a bending effort, only this last case has been studied because of it is the most possible cause of failure. This study does not completely model the behavior in real life of the limb due to the fact that carbon fiber is not an isomorphic material. However, the high security factor taken into account is expected to overcome this.

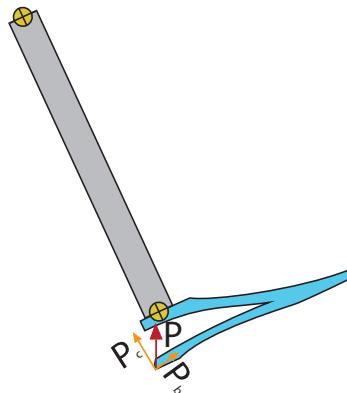


Figure 5.4: Impact force decomposed.

Profile study

The bending effort causes two types of problems: (1) the possible break in the supporting point and (2) the deformation suffered by the beam. The break might occur when the internal tensions created are over the ultimate effort in compression or tension of the selected material. This is modeled in equation 5.5 for symmetric sections and when a single torque M is being applied.

$$\sigma_{compression} = \sigma_{tension} = \frac{Mh_{CG}}{I_x} \quad (5.5)$$

⁴<http://www.easycarbonfibres.co.uk/#!/cured-carbon-fibre-products/>

Meanwhile the deformation in the extreme can be calculated with the equation 5.6 if the case is simplified to the one depicted in the Figure 5.5. This is, when the leg is completely stretched to its limits, which is when it will be subjected to the biggest stresses. Thus, the leg can be analyzed as a simple straight beam attached to a fixed point with no degrees of freedom, to which a force P (corresponding to P_b in 5.4) is being applied.

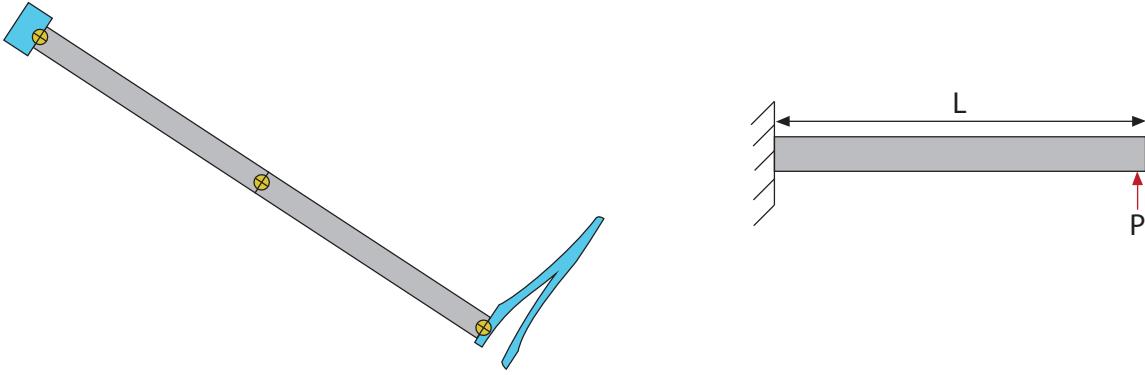


Figure 5.5: Simplified representation of the bending case.

$$y_L = \frac{Pz^2}{6EI}(3L - z) = \frac{PL^2}{6EI}(2L) = \frac{PL^2}{3EI} \quad (5.6)$$

For the selected profiles the equations that define the compression ($\sigma_{compression}$) and tension ($\sigma_{tension}$) efforts, along with the deformation in the direction of the applied force (y_L) are shown in the table 5.3, being:

1. h_{CG} : height of the center of gravity of the semi-half section from the geometrical center of the section.
2. E : Elastic module.

Torque calculation

For both the profile of the lower and the upper limbs, the torque generated by the impact force determined in the section 5.2.3, is calculated through equation 5.7. The torque differs in the limbs due to the fact that the applied on the upper limb is computed with the distance from the foot to the hip while the other one is only from the foot to the knee:

$$M_{lower\ limb} = F \cdot l_{link(s)\ length} \quad (5.7)$$

Final limb parameters

Using the criteria defined in sections 3.2 and 3.3 for sizes and materials, the presented formulas have been applied to all the profiles of the offered by the provider. The outputs

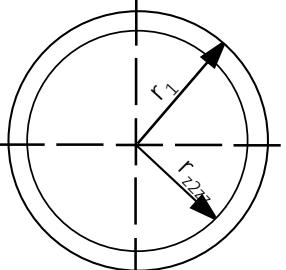
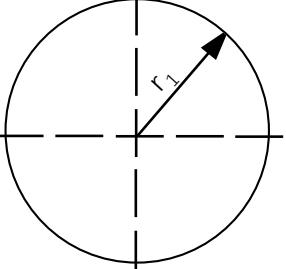
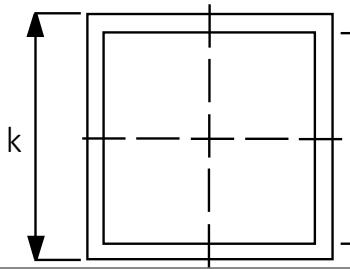
Section	I	h_{CG}	y_L	σ
	$\frac{\pi}{4}(r_2^4 - r_1^4)$	r_2	$\frac{4PL^2}{3E\pi(r_2^4 - r_1^4)}$	$\pi(r_2^4 - r_1^4)M$
	$\frac{\pi r_1^4}{4}$	r_1	$\frac{4PL^2}{3E\pi r_1^4}$	$\frac{4}{\pi r_1^3}M$
	$\frac{1}{12}(d^4 - k^4)$	$\frac{d}{2}$	$\frac{4PL^2}{E(d^4 - k^4)}$	$\frac{6d}{(d^4 - k^4)}$

Table 5.3: Stress and deformation analysis for the sections given.

of the iterative analyses in 5 and 4 are shown in the table 10.1. This final data is the used for the current stress study.

The profiles have been analyzed calculating the torque from 5.2.4 and then applying the section 5.2.4 for each iteration. The calculations of the last iteration are shown in the appendix .1. These results have been then approved or discarded based on a *maximum deformation* and *ultimate tension* requirements, and are shown in the table 5.4

Table 5.4: Profile selection for each limb.

Limb	Section	Dimensions
Lower limb	Tube	20 mm and 1 mm thickness
Upper limb	Tube	20 mm and 1 mm thickness

5.2.5 Ankle and knee joint mechanics

Several mechanical configurations were sketched as the first step of the iterations in the mechanical design of the joints. As previously explained, the hip joint turned out to have lower design constraints, since belt and compliance were not required, although the use of gears was a need. However, for the knee and ankle joints, the need of integrated

compliance and the torque transmission through a pulley yielded some restrictions that resulted in a final configuration as in 5.6.

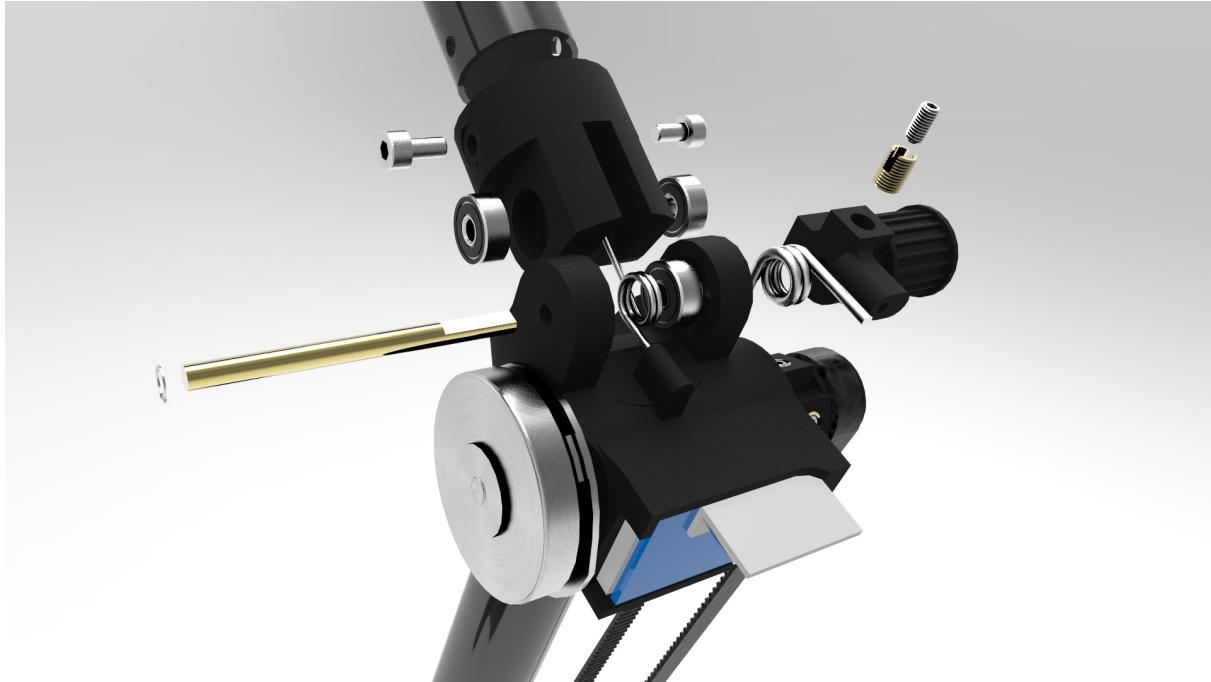


Figure 5.6: Knee joint mechanical implementation.

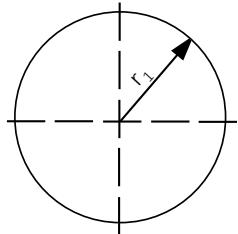
The configuration in 5.6 is equivalent for the ankles. Therefore, four bearings, a rod and the interfaces for the springs are needed and thus, discussed below.

Joints axes

The characteristics of the rod used as axis of the angular motion of the joint are studied here. It is also used, in the case of the knee and the ankle, as a support for the pulleys that transmit the power from the motor to the next link. Three mechanical efforts bound its design:

1. **Shear effort:** produced when an impact occurs and the internal efforts are transmitted through the rod between the two links that form the joint.
2. **Bending effort:** caused by the perpendicular force created by belt in tension on the edge of the rod.
3. **Torsion effort:** due to the pulley in the knee and the ankle. This effort is negligible because zero-friction bearings are supposed.

Shear analysis The maximum shear stress is found in the diameter of the cylinder ($y=0$) and is:



$$\begin{aligned}\gamma_{yz} &= \frac{Q_y M_x^A}{b(y) I_x} = \frac{Q}{r^2} \\ M_x^{A_{y=0}} &= \frac{\pi r_1^2}{2} \\ b(y=0) &= 2r_1 \\ I_x &= \frac{\pi r_1^4}{4}\end{aligned}\tag{5.8}$$

Given a tangent force Q , the shear stress can be calculated. If this is over the ultimate strength, the cylinder will break.

Bending The bending analysis follows the one carried out in the section 5.2.4 for a cylinder. The equivalent force in this case is given by the tension of the belts, mainly the initial (though there are other tensions that appear when the belts moves for example). Due to feasibility reasons and the lack of the appropriate measurements devices, some experimental tests trying different tensions and axes where carried out giving good results with a 3 mm rod or more.

Sizing The studies above have been tested for different diameters of rod starting from the smallest size given by the provider and increasing until both conditions are satisfied, due to the requirements of weight reduction. In case of using steel as material, the ultimate strength is supposed to be 250 MPa⁵. And for the case of a rod of 3 mm of diameter, both stress efforts are under the resulting restrictions. Thus, 3 mm rods were selected for both ankles and knees.

Bearings on the joints

The force calculated in section 5.2.3 has been used for sizing the bearings of the knee and the ankle. The bearing selected should be such that allowed dynamic loads of more than the impact force while being as small as possible to reduce additional weight on the frame. Besides, their internal diameter is defined by the rod diameter calculated in section 5.2.5.

An estimation of nominal life of the bearing can be done from the Dynamic Load Rating (C), the Dynamic Equivalent Load (P) and the Life Rime Coefficient for a Ball Bearing (p) (being $p=3$ for balls bearings). Equation 5.9 shows the nominal life of a ball bearing, which can be used in order to calculate the nominal life for a specific application. It is also worth mentioning that the Dynamic Equivalent Load (P) is divided by the number of bearings in which the force is spread.

$$L_{10} = \frac{10^6}{60n} \left(\frac{C}{P} \right)^p\tag{5.9}$$

The term L is the service life of a bearing (in number of hours or rpm), in normal conditions of speed and load, in which the bearing is working until fail by fatigue. Whilst L_{10} is

⁵https://en.wikipedia.org/wiki/A36_stee

based in a statistical model that is defined as the 90% of the bearing of the same type will withstand those loads for a longer time.

Springs dimensions

A big set of springs have been ordered in order to include them to the toolkit provided for experimentation, so that the trials detailed in 8.2 can be conducted. The criteria for the selection of the springs have been torsion model and 90°angle, ordering a left and right model of each since the mechanical interfaces on the robot are mirrored. Any of the acquired springs can be placed in any holder since the holes for inserting them have been designed with the diameter of the biggest one ordered plus a small clearance obtained from 7.1.1.

Exact calculation to obtain the stiffness of springs used for adding compliance to the joints of a legged machine have not been found in the literature. For research in a specific kind of gait, at a determined speed, the usual solution is to study real individuals and collect data from their analysis or empirically optimize the values through recurrent iterations until a good value is found. However, the fact that RuBi is not meant to be optimized for a determined speed or gait, but it is supposed to work in a wide range, prevented from conducting this kind of studies. Thus, the final values of stiffness were theoretically inferred from an adapted average of the implemented values for ankles in [43]. They correspond to the values of K in the middle of the range ordered. A list of the purchased springs can be found in appendix .4. The stiffest models have been selected to ensure the rigid behavior needed to implement a direct drive configuration in the actuation through a rough estimation of maximum torque requirements. Besides, the use of longer axes in the joints would allow the combination of springs in series for the SEA configuration, increasing the actual range of available values of K . This is suggested here as further work.

Implementation of elastic actuation

In section 3.4 the use of springs is justified in order to walk and run efficiently. An analysis of the different possibilities to include compliance in the robot is done in the same section, while Figure 3.3 shows the analyzed configurations and their advantages and disadvantages are further discussed. The final design has been conceived to allow for any of the four possible springs configurations shown in Figure 3.2 to be implemented in the knee and ankle joints. This is possible by making a system that allows to put elastic components in series and parallel that are easily replaceable.

SEA configuration In the section commented above, it is also determined the use of rotational springs for both series and parallel passive actuators. This gives a more constrained design that is later used as a feature. For example, in the section 5.2.1 is explained how the the pulley is integrated in a part that has the task of both being the place to put the spring and being the pulley. The Figure 5.9d shows this. This design shows the implementation of the SEA configurations that can be adjusted by changing the physical properties of the spring.

PEA configuration The parallel spring is implemented by adding a spring directly attached to both links of the joint. The problem of the parallel passive actuators is the need of loading the spring for movements in which is maybe not appropriated. As an example, the rest position of the parallel spring in the ankle could be placed when the foot is at 90 degrees with its consecutive link, so the robot can be stood up without applying any extra torque in the motors to keep balance. This would mean though to do an extra effort against the spring when taking off. In the same line of giving all the possibilities to the user with the compliance, a system that allows to change the rest position of the parallel spring has been implemented. This consists of one arm of the torsional spring attached to one of the links of the joint while the other arm can be fastened in different relative angles. The Figure 5.7 shows the transversal section of the joint component where the parallel spring can adopt different configurations.

The above explained design allows to implement a PEA configuration on the joints by just adding a spring and using an overdimensioned spring for the transmission from the pulley that can act as a DD.

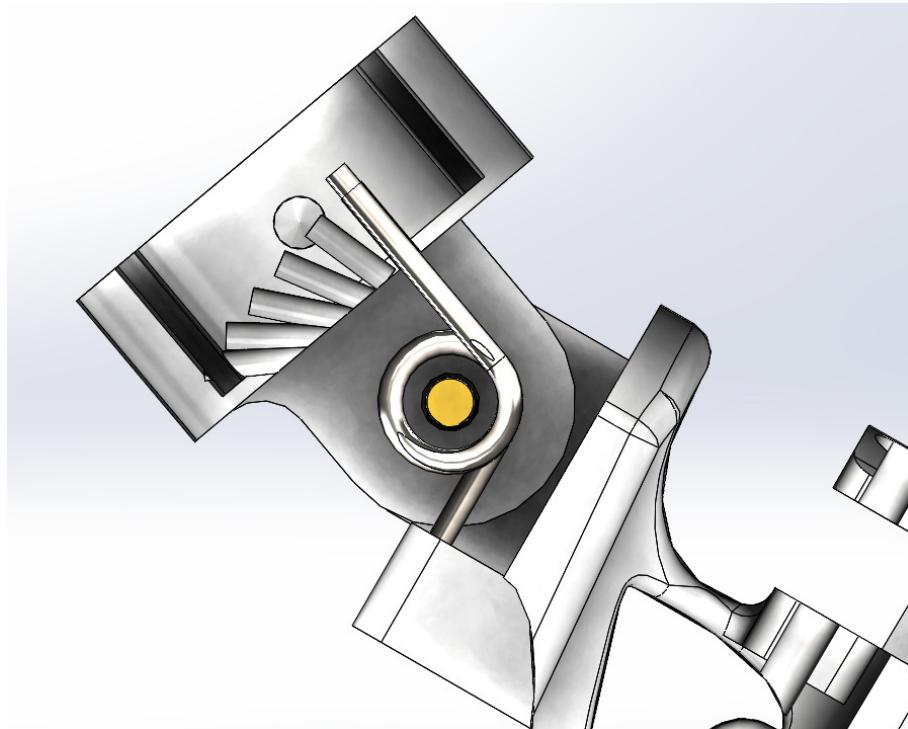


Figure 5.7: Transversal section of the parallel spring resting positions in the left ankle.

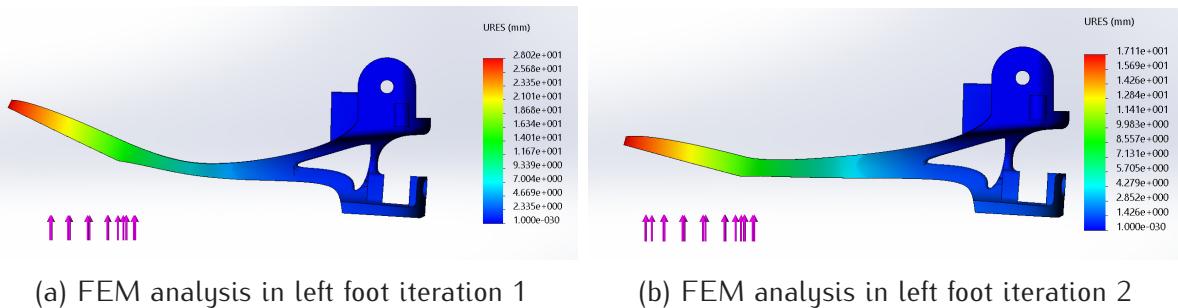
SEA + PEA configuration The possibility of using both, SEA and PEA, configurations at the same time is offered. This allows for the study of the SEA+PEA combination.

DD configuration Finally if no parallel spring is used and a spring stiff enough that does not add compliance to the system is placed in the SEA configuration, a DD transmission is achieved.

5.2.6 Finite Element Method (FEM)

In sections 5.2.4 and 5.2.5, a purely mathematical analysis has been carried out to calculate deformations and stresses on the limbs components. This has been possible due to the simplification of the problem and the simple geometries utilized. However, for more complex and critical parts a Finite Element Analysis (FEM) has been applied, in which the part is subdivided into small volumes and then analyzed individually.

As an example, in Figures 5.8a and 5.8b, a snapshot of the FEM in SolidWorks for the foot link can be seen. The initial design of the sole was subject to this analysis and reinforced in the specially sensitive areas in order to meet a minimum deformation criteria. However, all the mechanical designs tested here have always been also experimentally tried. This is due to the fact that FEM studies are fully valid only under the condition of isomorphism for the materials which, in the case of 3D printed parts like the shown in the figures, is not true. The FEM studies have been used more as a qualitative analysis rather than quantitative.



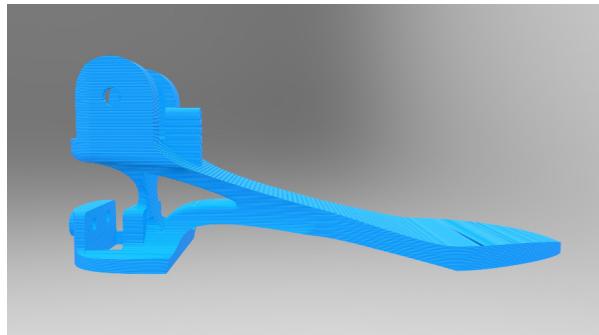
5.2.7 Computer-Aided Design (CAD)

After producing a final sketch for each component of the frame, the software SolidWorks was utilized to create their actual structure based on the required configuration, the desired placement of the sensors/actuators and wiring and the mechanical constraints calculated. In the absence of precise data about the physical requirements of a component, its volume has been increased to reinforce the resistance of the piece. When no mechanical or morphological constraints were to be applied, the criteria in CAD design has been mostly aesthetic, since the stile of the robot is worth considering, even if it does not influence its performance.

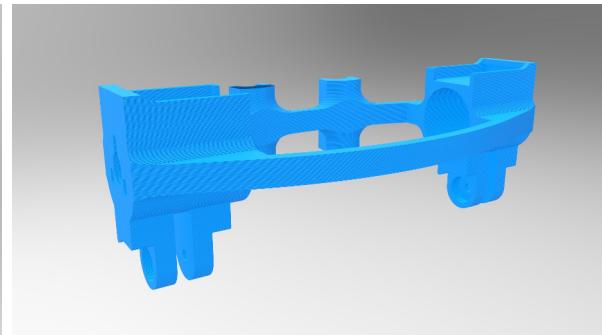
The CAD design has been carried out oriented to 3D-print manufacturing of the pieces, which adds new constraints in terms of resolution, size and materials available. Some corrections had to be applied to the tolerances of the internal holes as explained in the section 7.1.1 in order to avoid backslash. However, the advantages of employing 3D printers for the construction of the prototype compensate for these limitations.

The implementation of the CAD design can be considered partially parametric due to the fact that a file with some parameters has been used in several parts. This works so that in the reconstruction of the part, SolidWords reads the file and changes everything according to its content. This has eased the iterative process of optimization of the manufactured

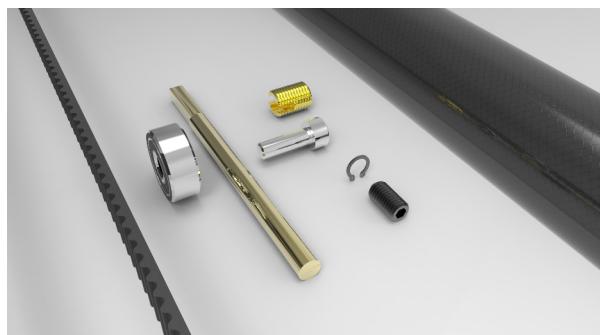
components. The values utilized are shown in the appendix 3. The results can be seen in the figures from 5.9a to 5.9j.



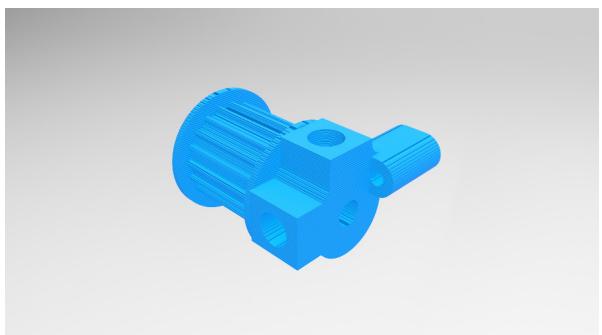
(a) Left foot



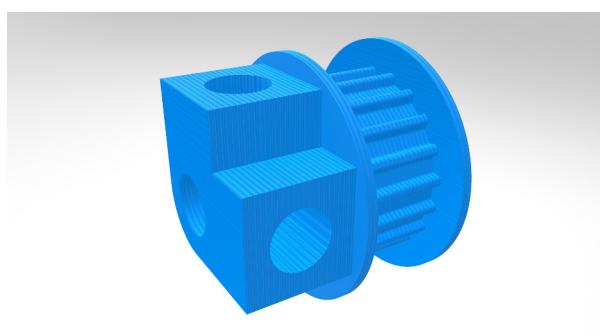
(b) Hip



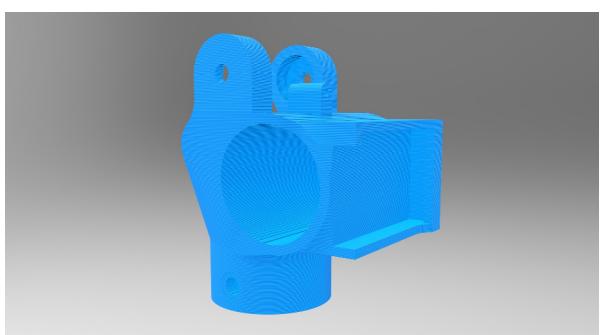
(c) Additional designed parts



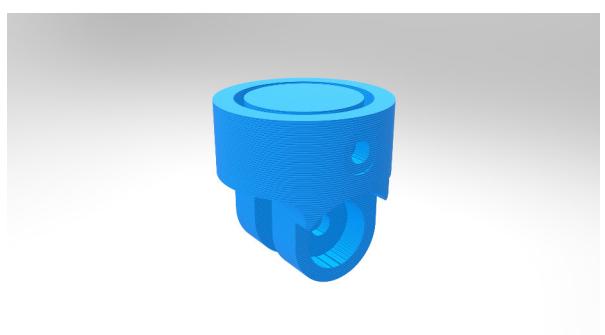
(d) Left ankle serial spring pulley



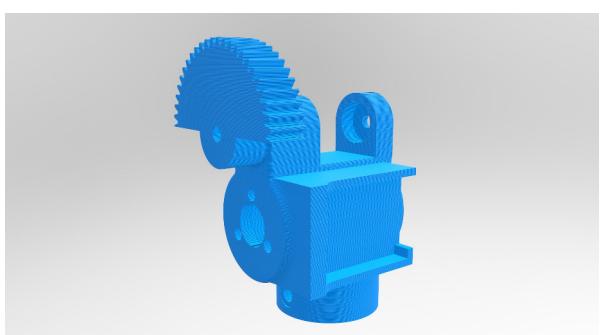
(e) Motor pulley



(f) Left lower knee

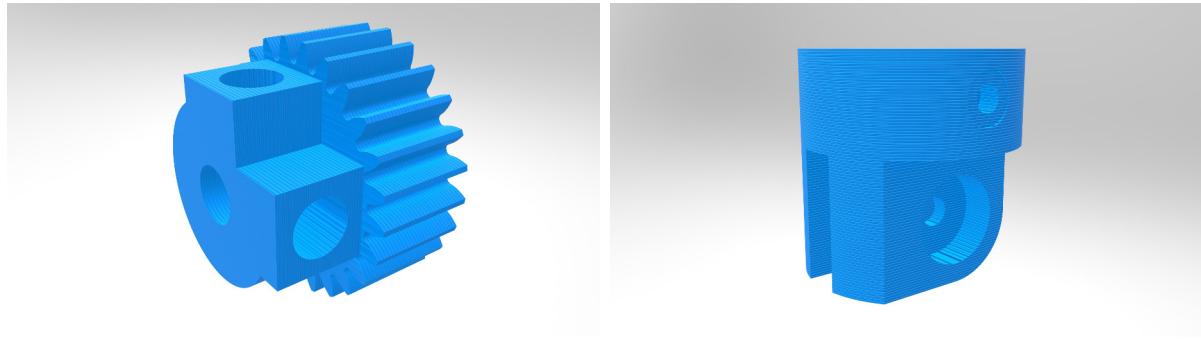


(g) Left upper ankle



(h) Left lower hip

The CAD program has the feature of, given the physical properties of the materials used, calculate the geometrical values of a component such as moments of inertia or CoM position. This has been used in the construction and use of the mathematical model in 4



(i) Hip's pinion

(j) Left upper knee

Figure 5.9: Main CAD designed components

and to size the motors, or in the simulation 6 to create the model as close to reality as possible. The table 10.1 contain the moments of inertia and total mass calculated by the CAD program. The moments of inertia are taken from the appropriate rotation axis of the link.

5.2.8 Mechanical limits of the joints

The implementation of mechanical limits for the joints obeys to two reasons:

1. **Calibration:** they can be used as starting, known positions for the relative encoders.
2. **Security:** a mechanical limit will restrict the movements and prevent any configuration non-natural or dangerous for the physical integrity of the robot.

The Figure 5.10a depicts how the mechanical limits of the hip (both left and right) are implemented in the upper part, comprising an angle of $90 + 60 = 150$ degrees. Figure 5.10b shows the upper part of the ankle and how the allowed movement of the foot has a range of $60 + 50 = 110$ degrees. The mechanical limits of the knee are obtained as a combination of the lower and upper components of the joint. These allow movements of $0 + 120 = 120$ degrees as shown in the figures 5.10c and 5.10d. The angles of the mechanical limits have been calculated according to the physical limitations in humans.

5.3 Software

5.3.1 Motivation

When approaching the task of designing a new robotic platform for the AI department at the Mærsk Mc-Kinney Møller Institute, the current development environment being utilized and its capabilities were studied as a first step. Nowadays, the simulation and control of the robots at the department is based on the LPZrobots [56] and Gorobots [35] packages.

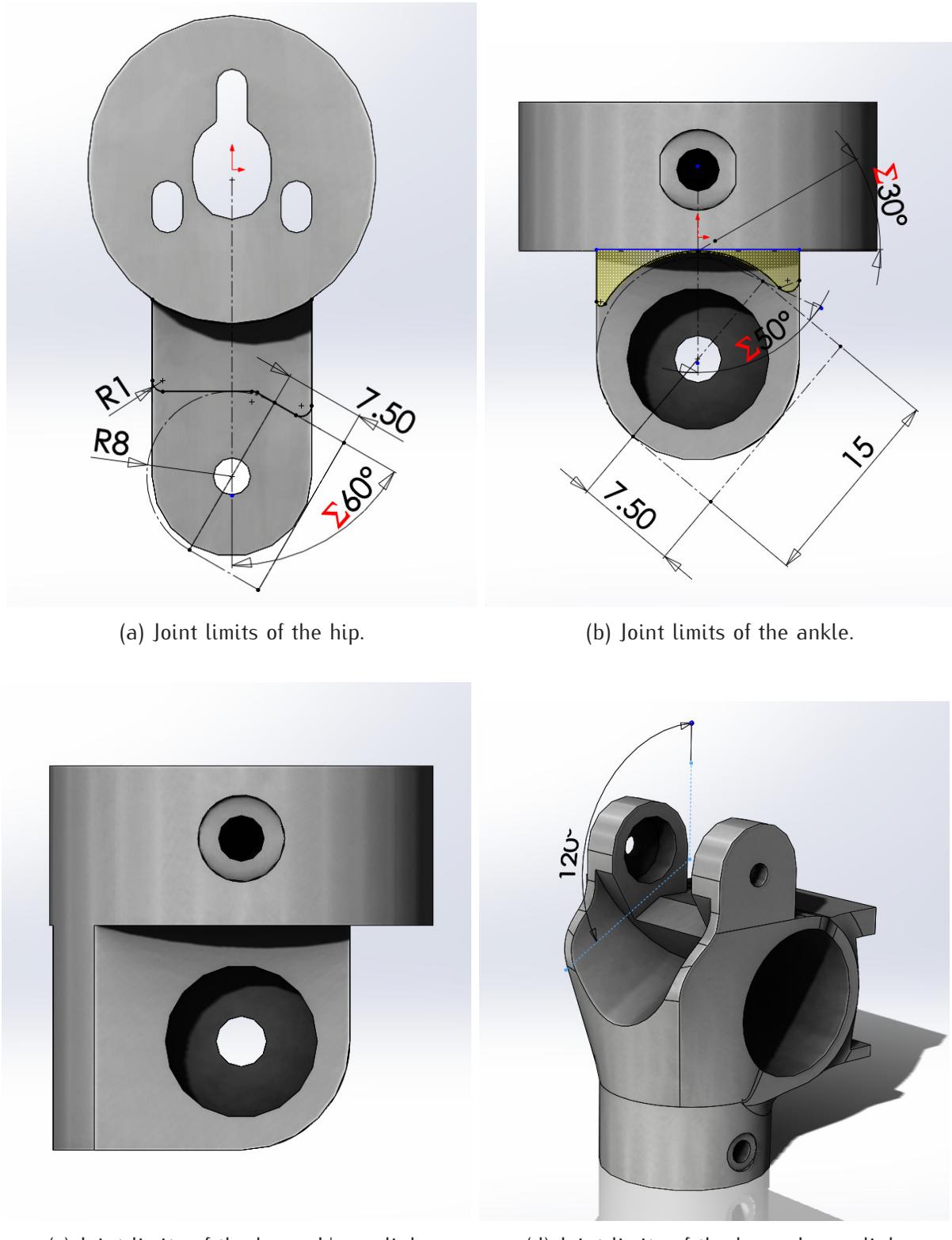


Figure 5.10: Figures of the joint limits implementation in RuBi.

These software tools provide both a simulation engine for robots built on ODE [50], OSG [51] and a framework for an easy implementation of controllers in both simulation and hardware. However, their specificity compared to other existing instruments collides with some of the core design ideas of the robot framework presented here, which are simplicity

of use and generalization. The above mentioned are the reasons why it was decided to migrate the development environment to ROS Jade [72] for the bipedal locomotion study framework of RuBy.

5.3.2 ROS Control

Within the Robot Operating System libraries, the ROS Control set of packages [58] contains several tools specially conceived for a generalized simple implementation of robot controllers. Besides, it standardizes the use of Gazebo [32] as a simulation environment with ROS by providing the necessary interfaces in conjunction with a simple plugin⁶. The architecture of the simulation, hardware, controllers and transmissions modules created through ROS control for both the Gazebo simulation environment and the a robot can be seen in Figure 5.11, as from the tutorial in [75]. The reader is referred to this online tutorial for a detailed explanation of the general features and the setup instructions of all the necessary modules discussed in this section.

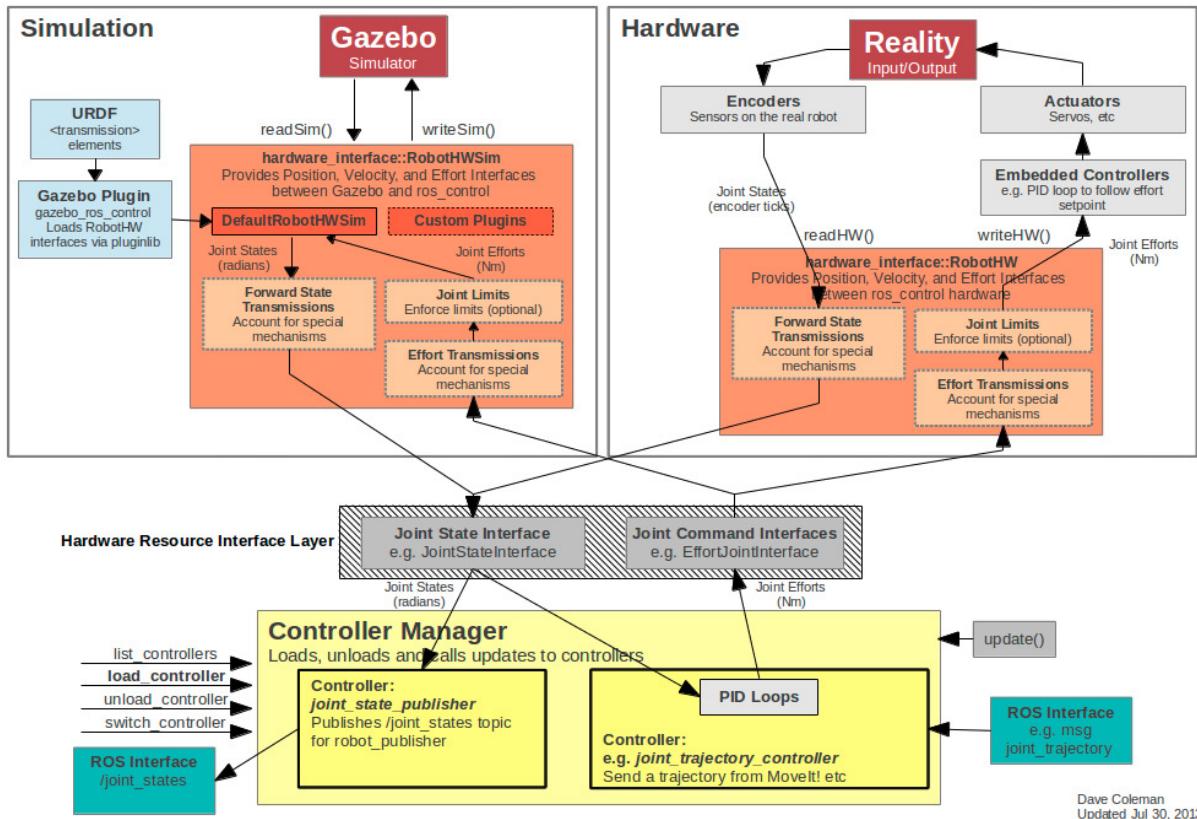


Figure 5.11: ROS Control and Gazebo data flow.

This architecture is meant to handle in real-time all the intermediary steps between the generation of joint commands for the robot by the custom controllers and their use in the platform, as well as for the feedback information flow in the opposite direction. The idea behind it is that for most of the actuator and sensor types in a robot joint, the basic

⁶The implementation of the simulation environment for RuBi in Gazebo and its adaption to ROS are discussed in 6.

commands and readings can be abstracted to a few kinds, being currently implemented position, velocity and effort instances. ROS control offers a set of controllers and transmissions for dealing with their transfer through the Hardware Resource Interface Layer, the Control Manager and the Controller modules in Figure 5.11. That leaves the "hardware interface" modules as the only part that needs to be defined by the user since they are platform dependent. The interface with the real robot is discussed in 5.3.3, while the one with the simulation is introduced in 6.

5.3.3 ROS Control-to-Hardware interface

The Locokit embedded electronics comes along a standard C library called LocoAPI designed for an easy interaction and use of the capabilities of its hardware, as presented in [41]. This library has been utilized to implement the basic hardware functionalities in RuBi without any modification. However, it can be easily edited or extended in order to add new ones if needed, which was considered a great advantage during the selection of the hardware platform.

The Locokit can be configured to offer a wireless interface between its main processor and an external computer creating and add-hoc WiFi network in the Gumstix. Besides, it offers a ready-to-use *C* application based on the LocoAPI to work as the server side when interfacing the hardware. Its setup and use instructions can be found in the Locokit documentation, supplied with the kit. This left the client side of the wireless connection as the one that needed to be created. An existing client application built as an "AbstractController" for the LPZrobots framework was already available. However, as explained before, to eliminate any dependency with LPZrobots and in order to use it within ROS Control, it was rewritten as an instantiation of "RobotHW" making use of the LocoKitInterface and ConnectionClass *C*++ classes provided.

The Locokit HW interface node

The resulting ROS node currently implements the functions listed in 5.3.3. They are fully operative and have been tested before assembling the actuators to the joints.

- Setup of wireless connection: connect to the TCP websocket created in the server side. It requires the IP and port on the server side.
- Registration of HW interface handlers for ROS Controller Manager: currently Effort commands and Joint State readings can be transmitted.
- Transmission of motor commands to motor boards in HW side: PWM signal values. It requires the physical addresses of all the motor boards connected.
- Reception of joint state readings from motor boards: encoder ticks for relative position.
- Real-time handling of information transfer from/to the Controller Manager.

The node can be extended with new standard/customized HW interface handlers and more functions from the "LocokitInterface" class for future applications. However, it lacks some capabilities that will be of relevance for an optimal functioning of the RuBi robot and that failed be implemented due to time constraints or lack of resources. The main ones are listed in 5.3.3.

- An initialization of the encoders to a predefined position set as zero in order to compute absolute position values.
- A new data flow channel to manage the sensory feedback information from non-built-in sensors, at the ground contact switches.
- If necessary, a mapping between the Effort command values received from the controller and the valid PWM signals sent to the motors.

5.3.4 Example controllers

From the *Controller manager* from ROS Control, different joint controllers can be handled. In the example controllers the position controller and the effort controllers are used individually for each joint. This joint controllers offer then a topic (e.g. /rubi/left_ankle_position) that the user can use to move the actuators. This are created from a unique package called *rubi_joint_controllers*. It is worth to say that the position controllers are implemented with a PID and that these values have been adjusted experimentally with the simulations for each joint.

Two type of controllers are given that show a different range of options to use with the robot. Both are gathered in a sole package called *rubi_controllers*, which gives a more tidy and resource-shared environment rather than having a package for each controller. This also makes really easy the deployment a new controller by removing all the creation process of a new package. In order to start a new piece of code this must be created and then added in the *CMakeLists.txt* from where also some examples are included.

The presented code follows the ROS conventions and the code style is the *Google style* offered by the clang-code-model. This creates a congruent workspace supervised by a git repository.

Two neuron controller

This first example controller shows how to:

1. Use GoRobots.
2. Make use of dynamic reconfigure.
3. Adapts its behavior depending on gazebo real time factor.

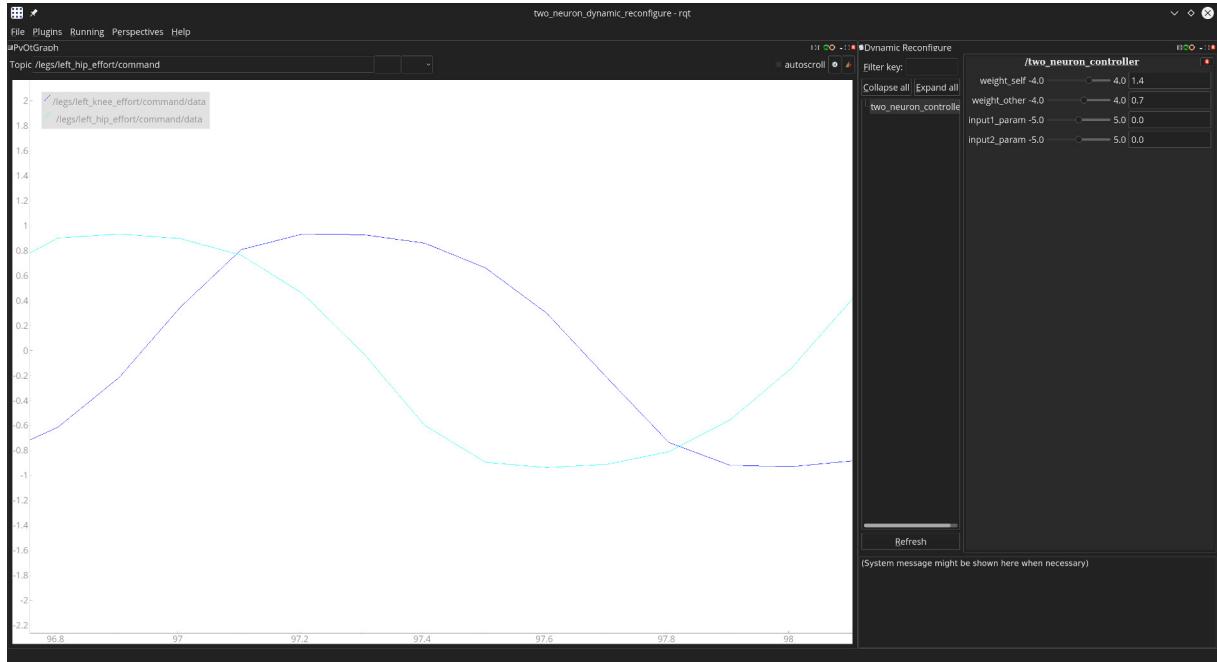


Figure 5.12: RQT workspace containing the CPG signals and the dynamic reconfigurable parameters.

For the first, an example of how to link C++ code from GoRobots is shown in the *CMake-Lists.txt*. This system is easier and more powerful than the current *Makefiles* currently used in GoRobots. The Artificial Neuronal Network (ANN) library is used to create a bi-neuronal network that creates the CPG signals of the actuators. Furthermore, the synaptic weights can be modified by making use of the ROS feature *Dynamic Reconfigurable Parameters*. These offer an interface in order to change, on the fly, the values of the created parameters. In the Figure 5.12, an RQT workspace containing the CPG signals and the dynamic reconfigurable parameters modifiers is depicted.

This are also used to change the behavior of the ANN in order to be adapted to changes in Gazebo. Gazebo uses can accelerate and decelerate the speed at which the time is passing through. This is adjusted with a *real time factor* that is read by the node and used to adjust the frequencies of the CPGs.

Impulse controller

Among other things this node shows how to:

1. Load and unload different joint controllers.
2. Give some wrappings for set of controllers (hoping position).
3. Offer services for jumping: given a file or given the values and impulse time.

This node implements some methods that enable it to change on-the-fly the controllers of each individual joint. Furthermore, three combinations of individual joint controllers are

given being these: (1) all in position mode, (2) all in effort mode and (3) left leg in position mode and right in effort mode. The last one is useful when hoping, a moment in which a leg must hold a position and the other keep pushing in order to jump.

Three different services are implemented that impulse the robot in the sake of rise it from the floor. The first two, are *impulse_one_leg* and *impulse_two_legs* which given a torque for each joint and an impulse time, it gives the possibility to jump either with one or two legs. Both services load the necessary joint controllers in order to achieve the desired movements. The third one offers sort of the same service but the parameters are read from a file instead. This is handy when combined with the dynamic controller developed in MatLab and explained in the section 4.4.

Chapter 6

Simulation

Within the goals of the RuBi framework is the creation of software environment whose last goal is to facilitate the user the test of new controllers. The process of testing new code in locomotion is usually slow due to the inherent problems of the hardware: maintenance, adjustments, bringing up the robot... Which breaks the workflow of the user and forces to spend a time that should be used in the actual development of the program. The RuBi framework includes a simulator and its simulation model in order to tackle this problem. The sections of this chapter include from a comparison of simulators to the explanation of how to start developing controllers, passing through the creation of the robot and other tools implemented like dimensional analysis or a rotational holder for the device. The instructions for the use of the simulation environment created are to be found in the manuals attached to this report.

6.1 Motivation

The robot presented in this thesis has been developed with security systems in both software and hardware meant to prolong its operation life and guarantee a safe use. Nevertheless, every device in real life is prone to suffer damages as part of their use or as a result of mistakes in its control translates into loss of economic and time resources for the user. A simulation environment as been implemented as a part of the development tools created with RuBi. Its architecture is such that the user can both work with the same algorithms in real and simulated life with a minimum reconfiguration.

While not being a complete imitation of the real life due to intrinsic limitations, it aims at offering a tool for qualitative research and testing of locomotion. As an example, neuronal networks or reinforcement learning algorithms, which learn from experience, can be developed faster in the simulator and then transferred to the real robot, reducing the workflow time and the need of resources of the user and justifying its use.

A simulator consists basically of two elements: a physical and a graphical engine. Whilst the first one is in charge of computing all the physical interactions that the agent is subjected to, the second one offers the visual experience. A congruent simulator with the

presented framework will satisfy two conditions:

1. The effort for testing among the simulation and real life must be minimized
2. The results must be as close to real life as possible. This includes multiphysics support (mechanical contacts, different aspects of tribology, wind, water...) but within a fair trade-off with computational load and speed.

6.2 Comparison of simulators

When choosing a simulation platform there are several factors to have into account. Three robotic simulators have been analyzed and compared. These are: LPZ Robots [56] (Figure 6.1b), V-Rep [57] (Figure 6.1a) and Gazebo [32] (Figure 6.1c). Some comparisons can be found in the literature as [49] or [66] however the predominant criteria has been their integration with ROS.

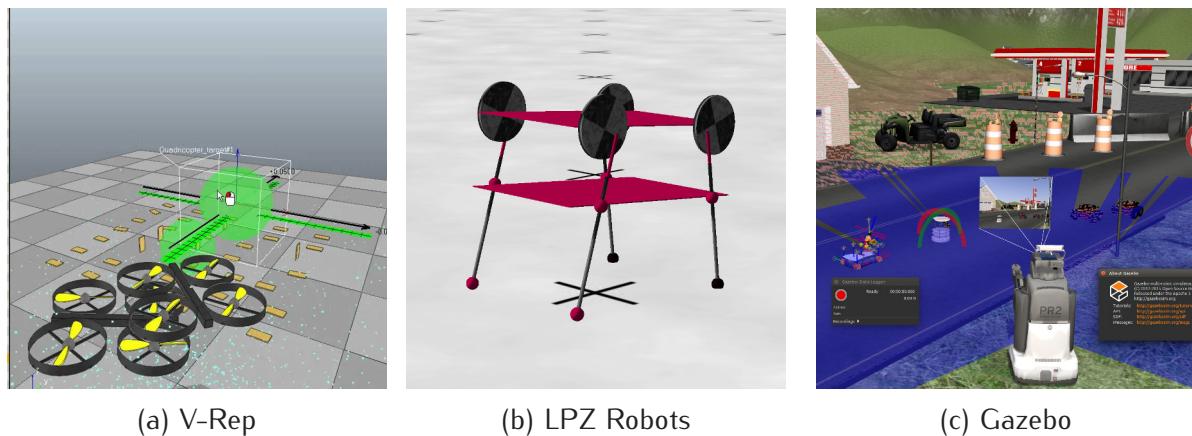


Figure 6.1: Simulation examples of the software analyzed.

The RuBi robot is mainly targeted to be used in the AI department at the Maersk Mc-Kinney Møller Institute, where the toolbox GoRobots is being developed. This is a set of tools, from a neuronal networks API to a genetic algorithm engine, written in C++ than is meant to be simulator-independent. However, in order to provide a more general and supported simulation and development environment for the user, ROS Jade [72] has been selected as the tool to use. The whole set of instruments provided in ROS along with its easy extendability give the opportunity to use the controllers and tools already developed in GoRobots. Despite the fact that the three simulators compared have a C++ interface that would allow to interface them with ROS, Gazebo is already fully supported and integrated in ROS, making it the logic choice. This reduces the learning curve of a new user and the installation process is easier due to it is included in the Open Source Robotics Foundation (OSRF) repositories.

Regarding the second condition, Gazebo has the feature of changing the physics engine in the beginning of the simulation, it has multiphysics support (fluids, electromagnetism...), it allows to load external geometries from STL or Collada and, the most important, has an

active community behind it providing support and documentation. These are the reasons for selecting Gazebo as the simulator for the RuBi platform. In [67], a comparison with the different physical engines is carried out. For the sake of simplicity the default one, Open Dynamics Engine (ODE), has been used, although the others have been tested.

6.3 Robot definition

As in May of 2016, ROS Jade ecosystem has different formats to describe a robot. These are SDF, URDF and Xacro. At some point they are all transformed to a unique sort of format but the decision of how to express the robot for the simulation is needed from the beginning.

1. **URDF**¹: is an open standard used in all the simulators mentioned or others like RobWork [29]. It allows to define all the properties of a single robot but lacks other which are important when simulating. It is mainly used for visual representations or schematic robot definitions.
2. **Xacro**²: is a parametric format that facilitates the writing of the URDF. It allows logic conditions like *if* or *for* and from ROS Jade virtually any python condition can be used. This code is then compiled into a URDF automatically which gives Xacro complete interoperability with URDF readers.
3. **SDF**³: adapted to the current requirements of the simulation environments. With it can be defined from *worlds* to air properties in the case of UAV simulations for instance.

Despite SDF contains more information, there are several tools in ROS Jade that make use of URDF. Two of them are RViz and ROS Control, explained in the section 5.3.2. As this last one is a pillar of the framework of Rubi, the decision of making use of URDF as format for describing the robot was taken. However, the robot developed have been written in Xacro due to the wide range of existing development tools.

6.4 Robot creation

One of the goals of the framework is to facilitate its continuity, thus this section is dedicated to explain how the robot was created, so that it can be modified in the future.

A robot is defined as a set of links joined. A link contains at least three blocks of information:

¹<http://wiki.ros.org/urdf>

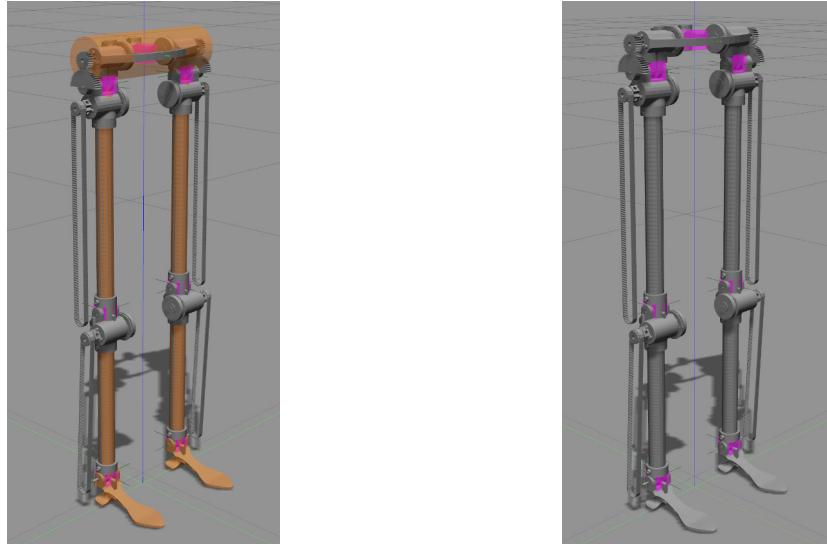
²<http://wiki.ros.org/xacro>

³<http://sdformat.org>

1. The collision model: used to calculate the collisions with other agents in the simulation.
2. The visual model: uniquely used for visual purposes.
3. The inertial information: defines physical properties like the mass and the moments of inertia of the link.

A trade-off must be found for the precision of the collision model between accuracy and speed. The more detailed is the model, the more accurate its simulation will become, but this implies a higher computational load and processing time. The general advice is to have two simulation models: one for visual purposes and the other one for collisions. In the case of the frame of RuBi the visual models are obtained directly by exporting the parts from SolidWorks, while the collision models are made out of primitive gazebo shapes (cubes, cylinders, spheres...) for the whole body except the feet. Since the robot is not meant to collide with obstacles due to its nature and setup, the possible collisions in the body are not a priority, then each link is represented as a cylinder of the same length of the CAD model and of a radius enough to cover the whole part. An example can be seen in the Figure 6.2a.

The moments of inertia of each individual link are obtained directly from SolidWorks. The CAD model includes the materials and thus the masses. From these, the program is able to calculate the moments of inertia. The Figure 6.2b shows the representation of these parameters in the simulation.



(a) Gazebo showing the visual model, the (b) Gazebo showing the visual model and the moments of inertia and the collision model.

Figure 6.2: Gazebo showing physical and visual properties of RuBi.

It is essential to make sure that the visual models and moments of inertia are exported referred to the correct coordinate systems. In the case of the ankle, for example, the STL file and the moments of inertia are calculated from the coordinate system whose origin is positioned in the middle of the rotational axis of the joint, with its Z axis attached to the

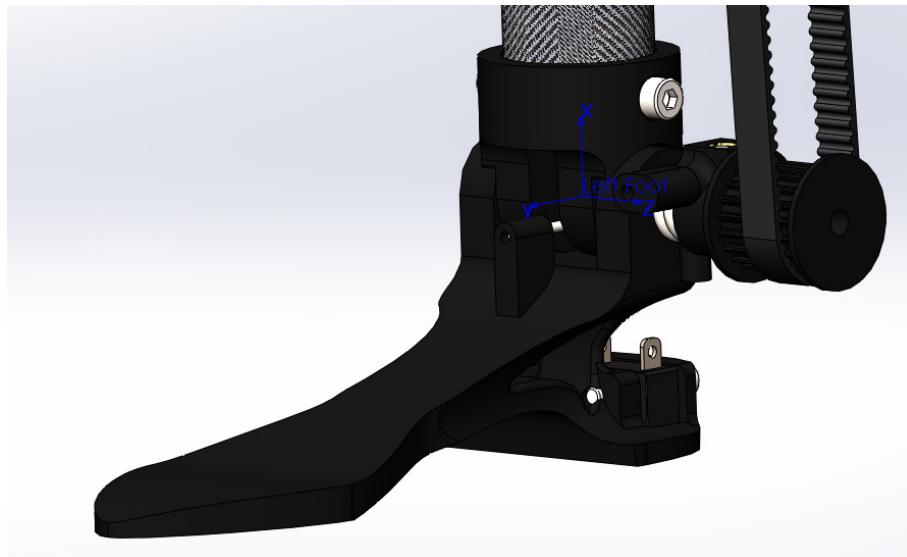


Figure 6.3: Coordinate system of the left ankle in SolidWorks.

rotational axis and Y pointing to the biggest extension of the part. This can be seen in the Figure 6.3.

Finally, in the Xacro file the information that lacks the URDF must be added in order to simulate with Gazebo. To do so, the following has to be added to the RuBi description:

1. **Plug-in for ROS Control:** Offers an interface for using ROS Control inside Gazebo.
2. **Contact sensors:** Simulates contact sensors as in the feet of the real robot.
3. **Friction coefficients:** Defines the friction between the feet and the floor.

6.5 Dimensional analysis

A problem faced when creating the robot was that some of the inertia moments obtained from SolidWorks were in the order of magnitude of 10^{-8} . Such small number cannot be handled by the physics engine correctly causing the robot model to be unstable under normal conditions. Three approaches were taken to fix this:

1. **Change the physics engine:** As commented in the simulation comparison 6.1, Gazebo supports multiple physics engines. Dart [28], Simbody [61], Bullet [23] and ODE [50], where tested giving the same unstable behavior. Thus, this method was discarded.
2. **Customize the moments of inertia:** This would entail rounding to zero the really small magnitudes, only leaving the ones over a threshold. As the simulation is a qualitative analysis and no quantitative, an exact match of the moments of inertia is not a strong requirement and therefore this option is left as a possibility.
3. **Dimensional analysis:** Based on [40], the size of the robot could be increased in a proportion big enough that would provide handleable values.

For option 3, dimensional analysis can be applied to correlate the physical properties of the original robot with its scaled replica. This problem could be formulated as: if the original robot has mass m , volume V and moment of inertia about an axis I , given a scale factor s , calculate the scaled replica m' , V' , I' depending exclusively on s .

In the equation 6.1, the generalized expression of the moment of inertia about an axis is shown.

$$I = \iiint_V \rho(u, v, w) |r^2| dx dy dz \quad (6.1)$$

If the equation 6.1 is taken as the moment of inertia from the first link, the scaled one would be the shown in the equation 6.2:

$$I' = \iiint_{V'} \rho(u, v, w)' |r'^2| dx' dy' dz' \quad (6.2)$$

Assuming a scale factor of s and as stated in the equations from 6.3 to 6.5:

$$dx = s dx' \quad (6.3)$$

$$dz = s dz' \quad (6.5)$$

$$dy = s dy' \quad (6.4)$$

Thus, it can be deduced that r increases proportionally with s as shown in the Figure 6.4 and presented in equation 6.6:

$$r = s r' \quad (6.6)$$

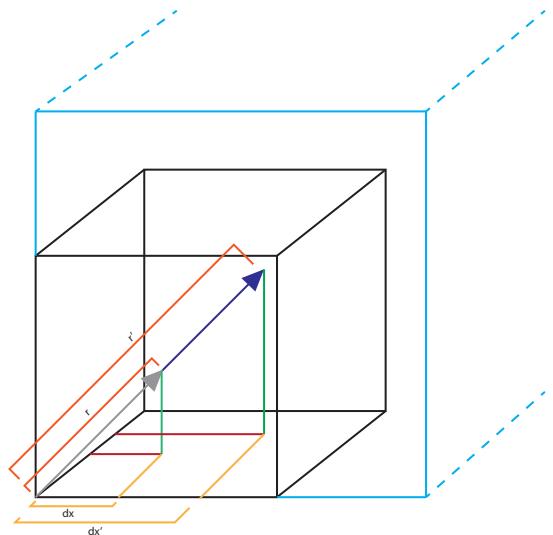


Figure 6.4: Representation of a minimal volumetric cube.

Assuming a constant density across all the scaled bodies, and particularizing for the moment of inertia of the CoG, the moment of inertia is given then by 6.7:

$$I'_{CG} = \iiint_{V'} \rho(u, v, w)' |r'^2| dx' dy' dz' = s^3 r'^2_{CG} m \quad (6.7)$$

A correlation can be obtained for the inertia then as shown in 6.8:

$$I' = \frac{s^3 r'^2_{CG} m}{r^2_{CG} m} I = s^5 I \quad (6.8)$$

And lately the mass can be correlated as in 6.9:

$$m' = V' \frac{m}{V} = s^3 m \quad (6.9)$$

As explained in 6.3, Xacro allows the use of mathematical expressions. This feature has been used to includes a set of variables in the definition of the robot which includes *scale* that is used to calculate *mass_scaled* and *inertia_scaled*. If the variable *scale* is changed, these parameters will then modify the values of all the links to make a coherently scalable robot definition.

6.6 Environment creation

Besides the robot, other elements can be added to the simulated world. From the Gazebo database, other robot models or parts (as a beer or a brick) can be inserted and used for any kind of experiments.

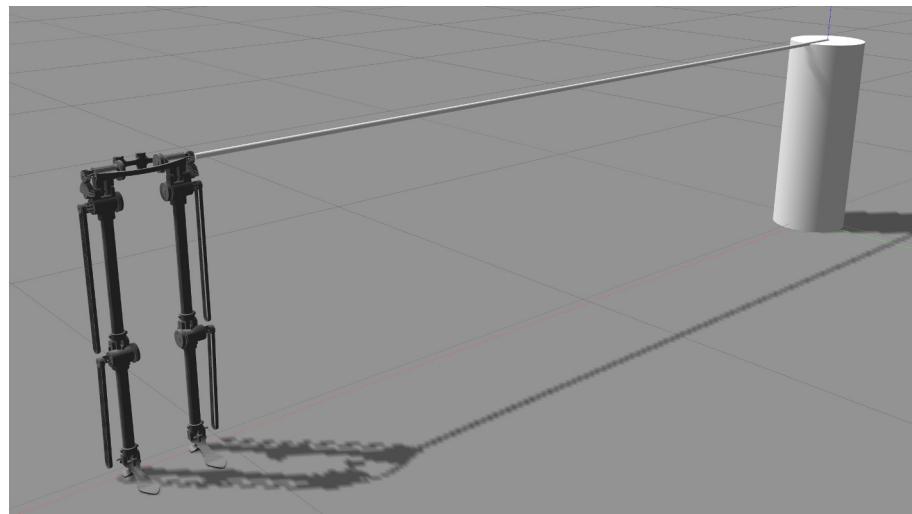


Figure 6.5: Rotational robot holder and robot.

In the final framework an extra tool is provided: a rotational robot holder. It is inspired in the DACbot simulation environment in LPZ Robots, being fully parametric and reducing the degrees of freedom of the robot to constraint its motion to the scenario under study. It can be seen in Figure 6.5.

Chapter 7

Construction process

This chapter is dedicated to explain the process of manufacturing and assembling the introduced designs. From the important role of emerging low-cost technologies like FFF 3D printing up to the creation of custom PCBs for reducing the weight of the robot, the following sections are meant to explain the process of acquisition and construction of all the components of RuBi presented in previous chapters. The implementation of the final prototype has always been led by criteria of security, constrained budget and feasibility for maintenance and futures improvements.

7.1 3D printing

The parts modeled for the project and shown in section 5.2.7, contain complex geometries that can only be achieved by 3D printing or complex machining. Due to its availability and low price, Fused Filament Fabrication (FFF) technology has been used to implement the parts although other methods were considered. The goals of this project requires the use of rapid and low-cost prototyping. Due to its intrinsic iterative design and the constrained timing, a technology that allows fast modifications in the design is required.

The additive manufacturing (AM) refers to the process of creating a geometry by adding material. Besides FFF, others technologies as Stereolithography (SLA) or Digital Light Processing (DLP) that make use of the light to solidify a photo-sensible liquid, generally give better accuracy but at the expense of a higher price. In this line, powder based AM like Selective laser melting (SLM) and Selective laser sintering (SLS), give also a higher detail although the post-processing and posterior mechanical properties tend to be lower.

Inside of the FFF, there are several materials from which a part can be printed of. Stand out polymers like Polylactide (PLA) and Acrylonitrile butadiene styrene (ABS), materials easily printable that have good physical properties. Its price can be considered inexpensive when comparing with the other AM technologies presented. The easy access to this technology, mainly powered by its low-cost tag, has caused an increase of the popularity of this kind of 3D printing giving as a result a big community. The outcome of this has been a wide variety of printable filaments with exotic properties: from non-flammable PLA

to carbon fiber reinforced parts with metal-like mechanical properties.

For this project FFF using PLA has been used. The reasons are feasibility and rapidness. The availability of multiple 3D printers and filaments made the selection of this technology straightforward. The 3D printer used is a M Prime One [45] with a 0.4 mm nozzle. PLA of two different colors was available, Blue and Black (the color may change the mechanical properties of the final printed part, due to different additives are used), from the provider BQ. All the parts have been printed at 0.2 mm layer height making use of free software slicers as Cura [26] and Slic3r [64]. The Figure 7.1 shows the gcode visualization of Simplify3D [62] of the left foot. One of the intrinsic problems of this technology is the

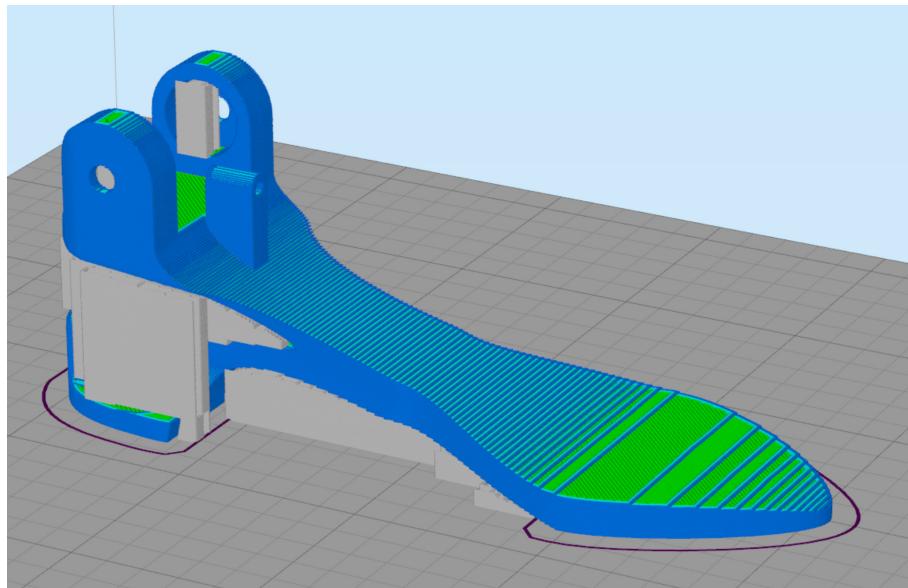


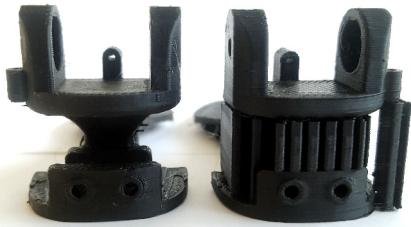
Figure 7.1: Gcode visualization from Simplify3D [62].

need of material support for some of the meshes. Due to its stratification technique, if one of the higher layers lacked a lower one to be supported on, the material would fall causing problems in final part. Nowadays, mostly all the slicers offer a material support solution that can vary but generally offering good results. The Figure 7.2a depicts two feet, one with material support and the other with it removed.

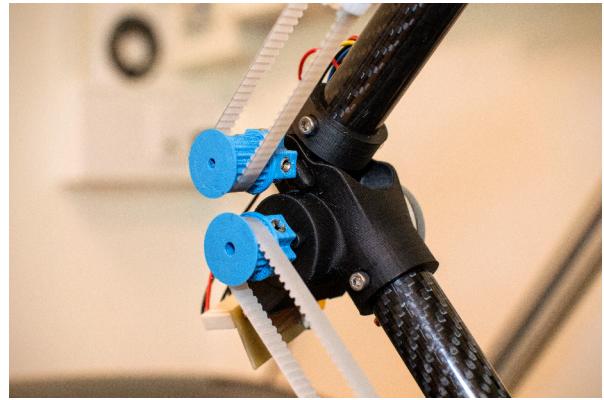
All the parts have been individually adjusted until the clearances have been as expected. This has been achieved by an recursive process and a mix between experimental tests and other tools like the presented in 7.1.1. In the Figure 7.2b a detail of actual parts 3D printed and assembled in the robot are shown.

7.1.1 Arc compensation in FFF

For the CAD models of the 3D printed parts, the clearances of the internal holes have been adjusted following [25]. The undersizing of internal holes is a common problem in this sort of technology due to the lack of information of the common-used exporting format: the STL. This only contains the 3D model expressed as a set of external triangles and its normal, which difficulties the correction of malformations inherent to this technology.



(a) Feet with and without material support.



(b) Detail of 3D printed parts assembled.

Figure 7.2: Photos of 3D printed parts.

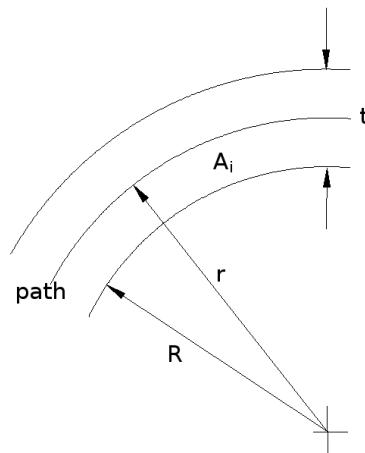


Figure 7.3: Technical representation of the generated arc when using FFF technology.

In the case of the Fused Filament Fabrication (FFF), the material is extruded equally in both sides of the arc, as shown in 7.3. However, in the side of the smaller curve, less material is needed. This correction can be calculated with the equation 7.1 being t the nozzle diameter, R the desired internal hole radius and r the corrected radius.

$$r = \frac{t + \sqrt{t^2 + 4R^2}}{2} \quad (7.1)$$

As an example, Klee suggests an internal hole of 4.4 mm in the case of the selected nuts [38]. Thus, the diameter in the CAD model has been adjusted for this data and a nozzle of 0.4 mm. The result is then show in 7.2.

$$d = 2r = \frac{t + \sqrt{t^2 + 4R^2}}{2} = 0.4 + \sqrt{0.4^2 + 4 * 2.2^2} = 4.81 \quad (7.2)$$

7.2 Machining

This section presents the process of machining followed for the non 3D-printed components that required some extra treatment during the assembly of the robot. The need of use of these parts is justified by the fact that the 3D printers used have a limited printing volume. Additionally, the mechanical properties given by the PLA did not satisfy all the requirements of all the parts. The rods of the joint axes, the beams used to create the holding structure or the carbon fiber tubes are some examples of these requirements of size and mechanical constraints.

The original carbon fiber tubes have been manually cut and drilled to give them their final configuration. Special attention has been paid to the symmetry of the frame during the construction following the design criteria established in 3.3. The joints axes rods have been also manually cut and filed according to the design. Besides, the holding structure mounted over the treadmill has been cut and assembled by hand as well. Knowing that this kind of operations is human error prone, the number of parts on the frame that needed to be manually produced has been tried to be minimized during the design. The drawings for such parts are included as appendices in .2.

7.3 PCBs and wiring

In order to reduce weight and inertias, 5.1.1 presented an extension board that interfaces between the motor and the BLDC electronics. With a 1.8 meter long wiring used to extend the connection, the purpose of the extension PCBs are to be mounted over the joints, close enough to the motors so that their original wiring can be used, as in Figure 7.4c, and transfer the signals to the motor board out of the structure, as in Figure 7.4a.

New Molex connectors have been soldered to the original boards in order to implement these extensions, as shown in Figure 7.4b within the final arrangement of the motor boards and the main processor board. A distinctive color has been chosen for the wiring of each motor just to ease the task of arranging the final configuration. Every motor board is labeled with the name of the joint it drives and matched to the appropriate address from the main processor by software, as explained in 5.3.3. This means that if a BLDC motor board is going to be changed, both its physical connection, label and internal address have to be changed accordingly.

All the off-board electronics has been placed over a small wooden structure to ease its use.

7.4 Providers

As explained in the management of the project section 1.3, all the components that could not be manufactured or were not available at the university facilities had to be ordered. The providers have been selected among the range offered by the Mærsk Mc-Kinney

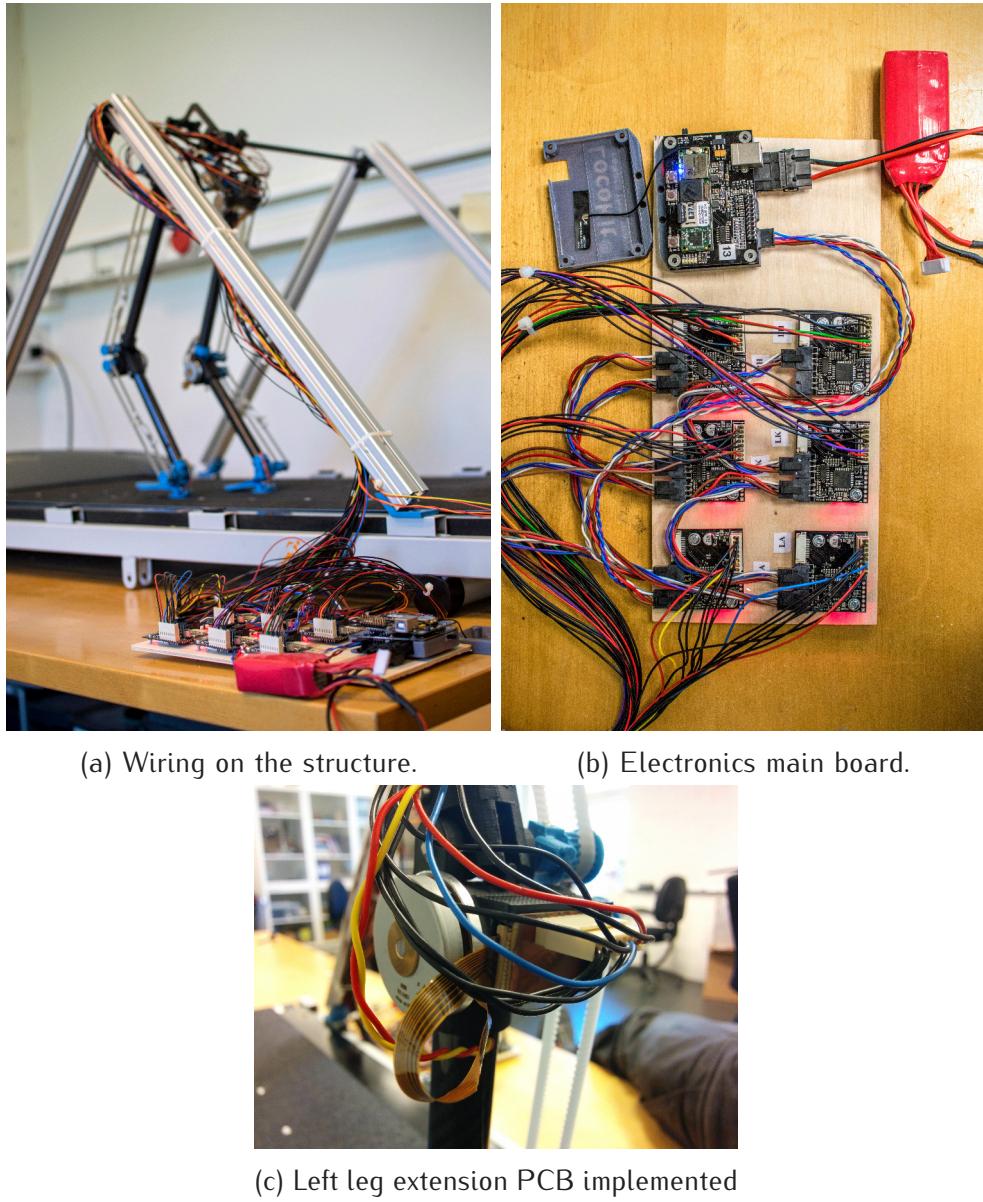


Figure 7.4: Final architecture of electronics.

Møller Institute. The main selection criteria of utilized external components has been standardization. Following the underlying principles of this project, only low-cost, standard pieces have been acquired. As an example, RuBi only needs three types of screws (only M3 metric) and one kind of bearing. A list with all the components bought and its providers is shown in the Table 9.1.

7.4.1 Components delivery

In the initial scheduling of the project, conducted at its very beginning, it was planned to have defined and ordered by the end of April all the necessary components that had to be bought. This was decided assuming that the delivery of the parts could take up to three weeks in a reasonable scenario. The acquisition of external components was

considered during the planning of the project as a strong time constraint, and the rest of the workflow was scheduled to adapt to this step. However, despite the fact that most of the parts were received within the first few weeks after ordering, the springs suffered a severe delay that prevented their scheduled assembly and produced a necessary reschedule of the project flow in order to adapt to this. This event has mainly impeded carrying out the devised experiments presented in 8 and some basic performance tests that were considered essential. This is further discussed in 10 and 11.

7.5 Assembly

Due to the fact that basically all the parts have been linked making use of 3D-printed parts, and all the tolerances of these have been adjusted individually, the assembly has been found easy and fast. The mounting process has proved to be rapid enough to change some of the parts, as the springs, within minutes. Furthermore, in the event of mechanical failure of any of the 3D-printed parts, its production and replacement is simple enough to reduce the maintenance time. This result of employing 3D-printed parts can be considered as one of the main advantages of the whole platform. The tension of the belts have been adjusted experimentally as explained in the section 5.2.1 using the zip ties installed. In Figure 7.5a, the complete robot assembled is shown while in 7.5b, a size comparison between the DACbot [6] and RuBi can be done.

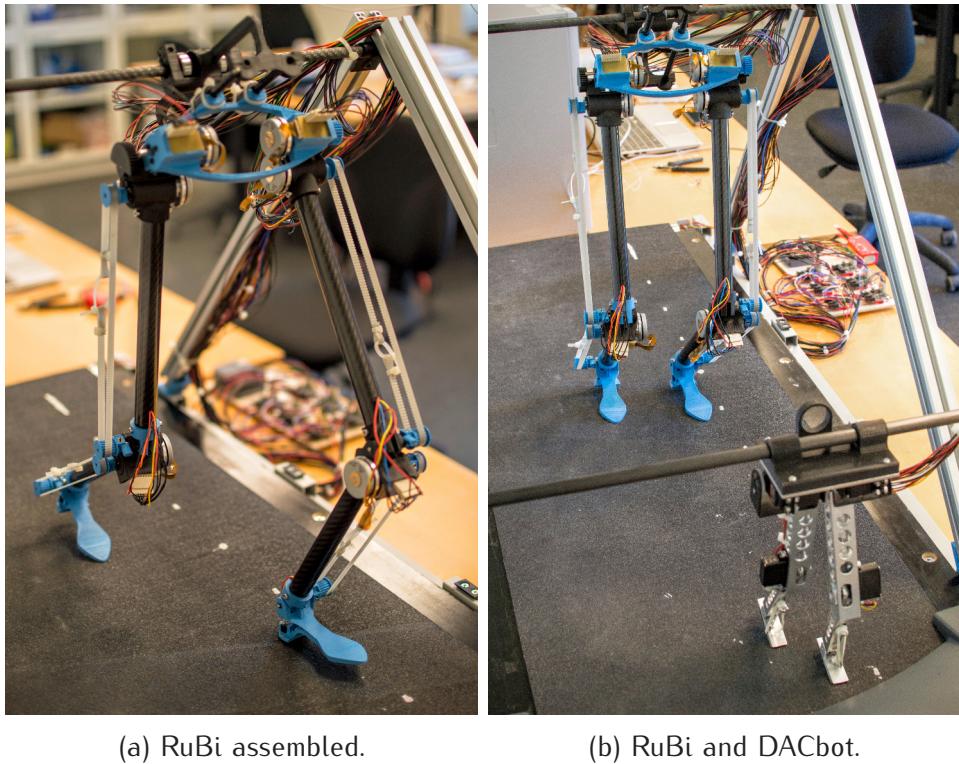


Figure 7.5: Actual photos of RuBi.

Chapter 8

Experimental framework

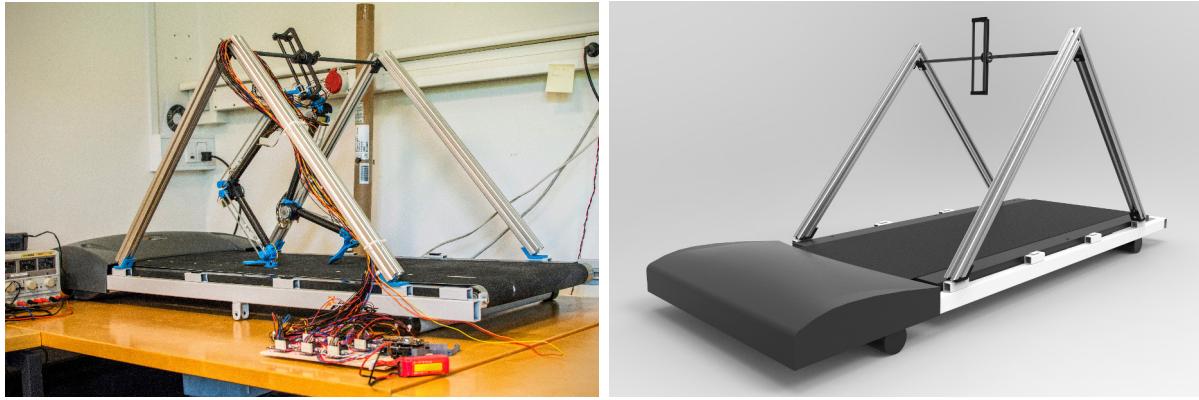
8.1 The test bench

During the simulation chapter 6, some tools were created along with the robot in order to carry out specific experiments. In that case the robot was restricted to a planar movement through the use of a rotational holder with an arm long enough to consider the trajectories as straight. This experimental setup was inspired by the already existing setup utilized with DACbot in the LPZ Robots simulator. However, in reality the rotational holder built for the DACbot robot was not an option for RuBi due to its dimensions bigger dimensions. A new designed was then implemented in conjunction with the existing treadmill used for locomotion experimentation. Thus, an equivalent structure that could allow the same restrictions as in the simulation over the treadmill was designed and constructed.

The result consists of a frame made out of a set of 3D printed parts and standard components, such as Bosch profiles and carbon fiber tubes. It is configured to permit RuBi only two degrees of freedom of combined rotation and vertical displacement on its sagittal plane. These two DoF, together with the relative motion provided by the treadmill allow for a full simulation of straight 2D motion. Besides, the rotation angle can be locked to a specific inclination to constraint the motion to vertical displacements. This feature was implemented to ease the experimentation on hopping dynamics explained in 8.2. In the figures 8.1a and 8.1b, a render and a photo of the real structure are presented.

8.2 Vertical jump experimentation

The theoretical framework presented in 4 to develop the formulas used to gauge the requirements of the actuators in 5.1.1 was meant to be tested empirically in order to assess its viability. However, the delays suffered in the delivery of some of the components for the prototype, explained in 7.4, together with the other eventualities described in previous chapters prevented from carrying out the devised experiments.



(a) RuBi on the assembled test bench.

(b) Original render of the test bench.

Figure 8.1: Photo and render of the test bench deployed.

Experiment 1

The jump analysis and dynamic model studied in 4.2 and 4.4 could be tested implementing springs stiff enough to model direct transmission and making the robot perform series of vertical jumps on one leg. To do so, the output of a fully-simulated jump cycle from the MATLAB controller discussed in 4.6 would be utilized as the input to the impulse controller created in 5.3.4, and tested in the real prototype on the test bench, constrained to only vertical motion. If the analyses were correct, there should be a correlation between the measured result of Δh for a given jump trajectory and the Δh expected.

Experiment 2

In order to evaluate the influence of the elastic actuators in the jump performance from an energy point of view, instead of using the equations presented in 4.5.2, an experimental approach is presented here. It is suggested the use of the virtual compliant leg model to shape the energetic contribution of each implemented spring to the energy storage capacity of the system. This method consists in analyzing the compliance of one leg as if it had only one linear spring from the hip joint to the toes, as shown in Figure 8.2a.

The energy stored by this linear spring can be decomposed as the sum of the quantities stored by each spring on the leg, as shown in 8.1. Where Δl_{eq} would be the changes in the length of the virtual spring and the constants of the springs are known. However, the changes in the state of the spring can be hard to calculate or measure empirically. An study of the different stages that the energy in the system goes through while dropping the robot on one leg from a known height is suggested instead:

1. If the robot is dropped from a know altitude h_{drop} in a certain configuration, that could be considered the equilibrium position of the virtual spring, shown in 8.2a. At that point, the system would contain a potential energy characterized by the initial height $E_{pot}(h_{drop})$.
2. The instant after landing for a maximum compression of the springs, right before

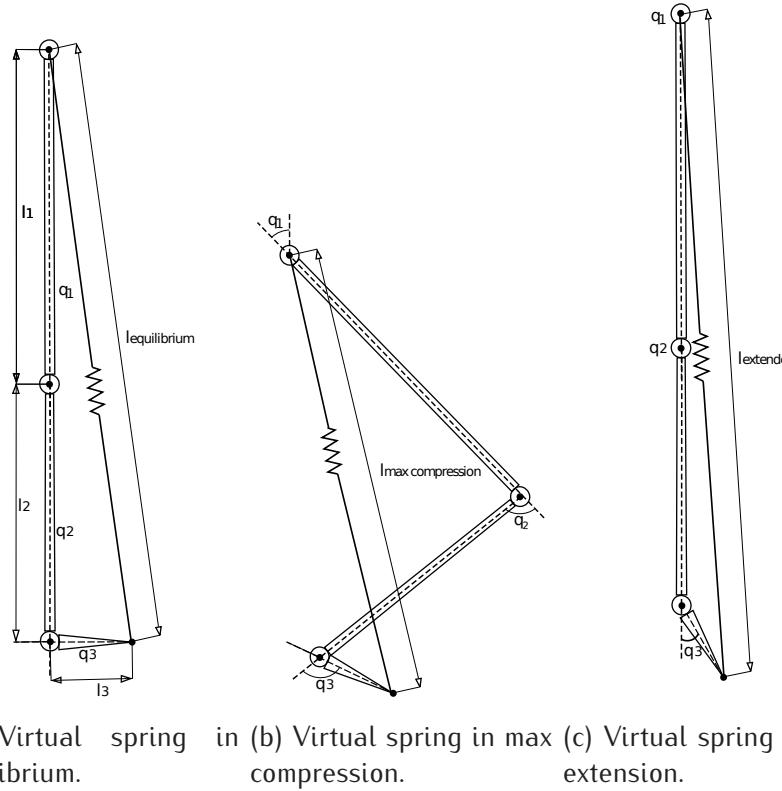


Figure 8.2: Virtual spring representation in different states.

starting the launching reaction caused by the energy stored, is seen as the maximum compression instant, depicted in 8.2b. Here, all the potential energy has been stored on the springs or lost in the ground contact and trough system losses very difficult to model or measure, E_{loss1} as internal frictions. This change in the energy state in the system is represented as in 8.2.

3. Finally, the take-off configuration, in 8.2c, would be the one reached once the springs have released the energy and the maximum acceleration upwards is reached. Once the robot has reached its maximum height $h_{rebound}$ all the energy left in the system, E_{eq} , has been converted to potential energy again plus other losses, as in equation 8.2.

$$E_{eq} = \frac{1}{2} K_{eq} \Delta l_{eq}^2 = \frac{1}{2} K_{knee} \Delta \theta_{knee}^2 + \frac{1}{2} K_{ankle} \Delta \theta_{ankle}^2 \quad (8.1)$$

$$\begin{aligned} E_{pot}(h_{drop}) &= E_{eq} + E_{loss1} \\ E_{eq} &= mgh_{rebound} + E_{loss2} \end{aligned} \quad (8.2)$$

To reduce energy losses in non-vertical displacements, the robot trajectory should be physically constrained. Within the presented theoretical frame, the energy stored by the virtual spring could be calculated simply by measuring h_{drop} and $h_{rebound}$. However, the right side of equation 8.1 cannot be decoupled. Instead, the configuration and number of springs for several series of the same experiment could be changed, and their influence

on the obtained E_{eq} compared to assess their contribution to the overall energy storage capacity of the leg. A high-speed vision system to provide the height measurements for the experiment, following the model presented in [20], is put forward here for consideration.

It suggested the use of a position controller for the joints in order to fix the joints positions and create a rigid structure of RuBi. This kind of controller has not been implemented on the top of the current software interface. However, its fully implemented in the electronics firmware and can be easily created in ROS Control through the standard HW interface class.

8.3 Simulation experiments

In the section 5.3.4, the developed example controllers were presented. Among them, the impulse controller was created to provide a set of custom peak torques to the joints and generate impulse motion on the frame. Although the resulting impulses are not controlled in anyway, they permit a fast look to the order of magnitude of the forces necessary in order to jump. The controller has only been tested in the simulation and in Figure 8.3, RuBi is shown jumping as a result of the torques applied by the controller. While a real conclusion of the jumping capabilities of the robot can only be achieved by tests in the real robot, the simulation proves to be useful for conceptual testing of the controllers.

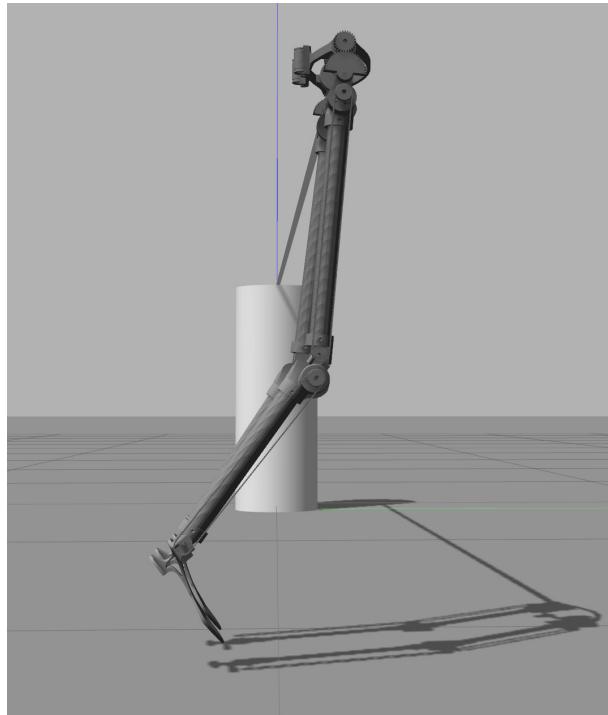


Figure 8.3: Screenshots of the robot taken when jumping by using the impulse controller.

Chapter 9

Economical analysis

In this section the economical aspects of the thesis are detailed. The labor cost was neglected as it is difficult to measure the dedicated hours to the project in both sides, the students and the supervisors. In the same line, the equipment and software costs have not been included due to the uncertainty when calculating the associated cost to the project of, for example, the computers used in the development of the project. Thus, only the material cost is shown in the Table 9.1 showing a total of 22,772 DKK including all the elements used for the project, not the ordered ones.

Some of the components where available, like the LocoKit, and therefore were not bought. Others like the tools or the laptops are not added as they are supposed to be not project-specific. Analyzing the budget, the bulk of the money goes into the LocoKit, that includes the electronics and the Maxon motors which are the most expensive part of RuBi. However the rest of the components show to be affordable what makes its replication and reparation simple from an economical point of view.

Table 9.1: Material cost.

Material Cost			
Item	Quantity	Unitary cost [DKK]	Cost [DKK]
20mm(18mm) Carbon Fibre Tube	1	192.51	192.51
10mm(8mm) Carbon Fibre	1	99.21	99.21
8mm(6mm) Carbon Fibre Tube	1	91.97	91.97
Kugleleje SS623-2RS	26	20.67	537.42
Contitech Tandrem	2	137.18	274.36
Låsering	1	28.48	28.48
Rustfrit	2	192.07	384.14
Skrue M3 x 6mm	1	104.82	104.82
Skrue M3 x 3mm	1	96.61	96.61
Stiverprofil	4	91.97	367.88
RS Pro Kugleleje	2	16.44	32.88
SKF Lineært kugleleje	2	116.89	233.78
Fastgørelse og forbindelsesled	1	62.76	62.76
RS Pro Glat bøsning OB368	1	54.06	54.06
Skrue M6 x 10mm	1	34.61	34.61
Torsion springs	1	1,454	1,454
3D printer filament	1	140.00	140.00
LocoKit	1	18,592.00	18,592.00
Total [DKK]			22,772.19

Chapter 10

Results

The main result of this project-oriented thesis is the first prototype of the bipedal platform for human-like locomotion studies RuBi, and its development and study environment. The work carried out for the last four months has comprised from the origin of the idea and its initial conceptualization to the last steps taken on its mechatronics construction, the development of its control and electronics architecture, its simulation model and environment and the devise of the tests it should have been subject to in the presence of more time. The outputs of the whole workflow presented in this report are summarized in this chapter for an overall view of the results, although they have been discussed more in detail in their respective chapters.

10.1 Morphological properties of RuBi

The geometrical and inertial dimensions of the final prototype of RuBi can be found in table 10.1. The geometry of the robot has been inspired in the lower limbs of a male person in order to facilitate human-like motion for research in this field. However its inertial properties have been minimized in order to reduce power requirements and their inherent economical aspects (a more powerful powertrain would entail greater costs). From a mechanical perspective, this has been translated into the application of light but resistant materials, while striving for low cost, standardization and easy maintenance, achieving a low weight and reduced inertias, as can be seen in table 10.1. The use of 3D-printing as the manufacturing method for most of the custom components designed has contributed to make the structure lighter as well as to speed up the prototyping process and make the final structure easier to repair and modify.

A comparison between other two-dimensional bipedal robots of similar morphology to RuBi can be seen in 10.1. It can be easily appreciated that the overall length-weight ratio of RuBi is significantly better than more developed models as the Phides [27] or BioBiped [14], even considering that they have an on-board controller and all this implies.

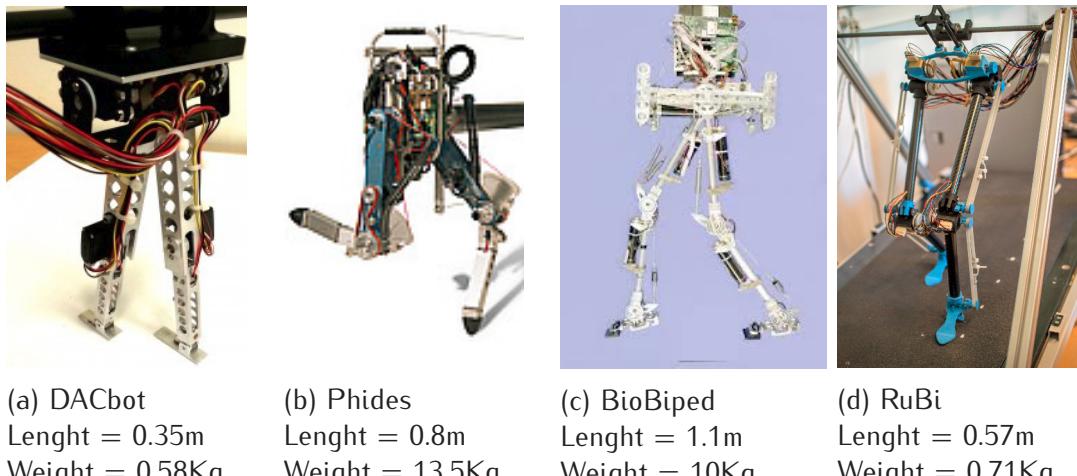


Figure 10.1: Length-weight comparative with existing bipeds.

10.2 Actuation and sensory feedback

RuBi has six actuated joints for planar motion. For each of them, a fully functional and extendable embedded electronics system based on the Locokit hardware has been implemented as the spinal cord of the robot, together with a software interface able to handle the information flow with the robot. It provides access to the angular information of each joint in real time and the control of their individual torque commands, all through a standard ROS interface that can be used as well for controlling the simulation model. As an example, the readings from hip, knee and ankle encoders for a random movement on one leg are shown in Figure 10.2.

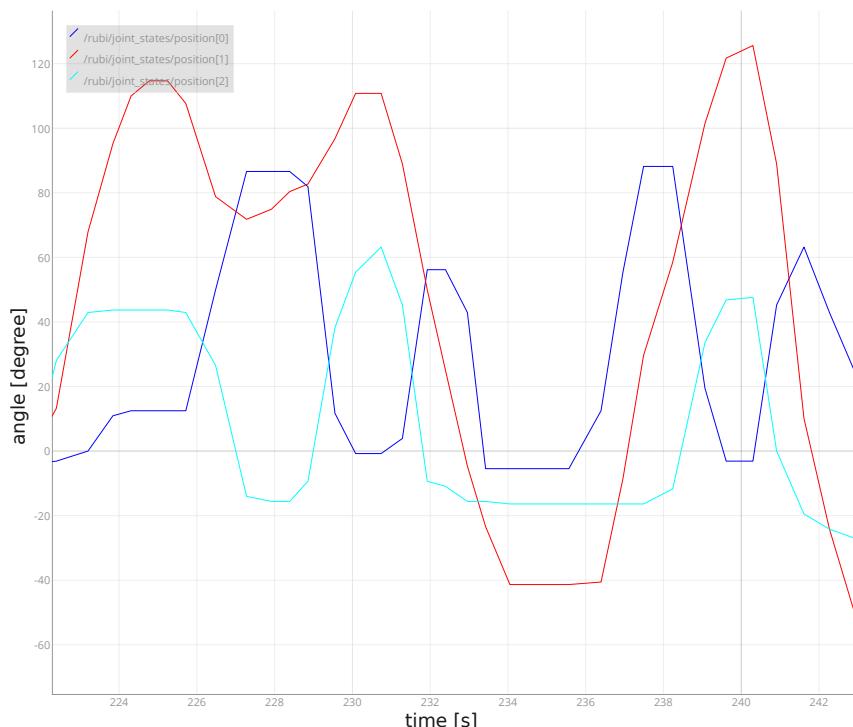


Figure 10.2: Measurements obtained from real robot.

Limb	Length [m]	Mass [Kg]	Moments of inertia [Kg · m ²]		
Foot	0.095	0.021	2.2e-005	-9.16e-006	0
			-9.16e-006	9.05e-006	0
			0	0	2.88e-005
Lower limb	0.115	0.2	0.000699	2.44e-005	0
			2.44e-005	1.85e-005	-1.82e-005
			0	-1.82e-005	0.000694
Upper limb	0.123	0.2	0.00119	2.89e-006	-1.36e-008
			2.89e-006	1.69e-005	-1.78e-005
			-1.36e-008	-1.78e-005	0.00118
Hip	0.150	0.156	0.000303	0	1.44e-008
			0	2.78e-005	0
			1.44e-008	0	0.000314
Total		0.707			

Table 10.1: Physical properties of the limbs obtained from CAD program.

The six actuators on the robot has been individually tested once on-board although detached of the transmissions due to the delay on the delivery of the springs explained. Its selection has been based on the calculations introduced in Algorithm 1 detailed in 5.1.1, that have made use of the dynamic model of the robot constructed.

All the control hardware has been placed off-board to reduce weight on the frame.

Contact sensors have been placed on the heels of the robot to offer the possibility of monitoring ground contact during the locomotion. However, they have not been connected to the electronics frame due to unforeseen issues with the hardware frame, that could be easily overcome with simple additional circuitry and some more time.

10.3 RuBi dynamic model and actuators calculation

The MATLAB scripts containing the implementation and results of the dynamic model of the robot and the calculus for the parameters of the actuators can be found in the source folder attached to this thesis. The final model has been considered a fair trade-off between simplicity and accuracy, since the addition of more terms hard to model as frictions or backlash in the transmissions would not have contributed additional value.

Section 4.6 has presented a possible intrinsic use of the dynamic model of the robot, which is to conduct torque control on the robot through its real-time computation. However, the application of the model as a controller has not been fully developed and it is just left in

this report as a proof of concept and to suggest its potential use.

10.4 Framework for experimentation

A novelty of RuBi is the possibility of implementing four different elastic actuation configurations in each knee and ankle joint. SEA, PEA, SEA+PEA and DD configurations can be utilized for the transmission of the motion to the joints. This converts RuBi into an ideal platform to study the influence of the compliance configuration on the energy storage capacity of a bipedal platform or on the power peak consumption of the joints, for instance.

For facilitating the conduction of locomotion experiments, a test bench that allows a fully controlled 2D legged motion for the robot has been built on the top of a treadmill. It can be easily modified to be used for hopping or jumping motion experimentation. Some experiments devised for testing the prototype are suggested in 8 although not carried out due to time constraints and delays suffered.

10.5 Simulation environment

A simple and extendable simulation environment together with a robot model have been created in Gazebo. The physical properties of the robot and its visual model have been exported from its design in SolidWorks for maximum resemblance with the real device. A ROS interface has been created for the simulation environment, together with two simple controllers that have been successfully tested (see 8). They have been used also to prove the simplicity of the development and testing of control algorithms with the simulation model. In Figure 10.3 a sequence of snapshots depicts RuBi performing an unconstrained jump as a result of the use of the impulse controller in the simulation. It has been used to demonstrate that the physical characteristics of the structure are capable of handle the motion implied in a jump.

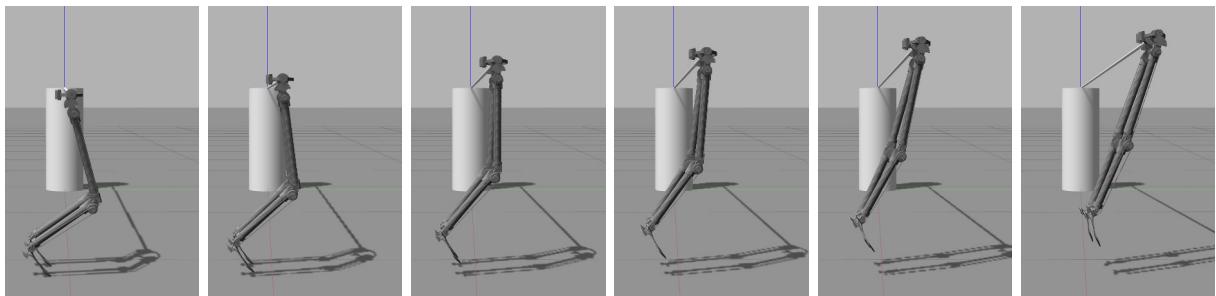


Figure 10.3: Snapshots of RuBi performing an unconstrained jump.

However, the problems of the physics engine to handle robots of the size of RuBi led to the need of truncating some of its too small physical properties in order to be able to simulate real behaviors. Hence, the simulation model can be used to qualitatively

test the functionality of controllers, but their outputs will have to be adapted to the real environment until the simulation model is further improved.

10.6 Mechanical implementation

From the perspective of the mechanical assembly all the steps on the planned procedure have been finally taken with the arrival and mounting of the torsion springs used for the elastic actuation in knee and ankle joints. Despite their unexpected delay, the models T048-090-218 and T051-090-234 have been built in SEA configuration of the prototype for testing purposes. The main goal was to test the functionality of the springs holders on the interfaces. The complete transmission has been manually tried yielding the expected results. Therefore the last step should be to tune the tension on the belts to guarantee the best torque transfer.

Chapter 11

Discussion

The purpose of the project was the construction of a robust, low-cost and reconfigurable bipedal platform for human-like walking and running locomotion generation and control studies. Once the results of the process are collected, they deserve an evaluation in terms of the goals aimed with the thesis and the development of events within the past months.

11.1 The first prototype of RuBi

The core of the devised framework for locomotion studies was the biped robot RuBi. The design and construction of the first prototype has been fully achieved as planned despite the eventualities faced. Its theoretical sensory/motor and mechanics features accomplish most of the targeted requirements for a robot of its characteristics, inspired in similar, existing models in the literature and the authors own intuition. However, the next chapter details some tasks left as further work considered necessary for a complete success on the implementation of the fully functional prototype has conceived.

The main deficiency on the development process remains the absence of empirical data to corroborate the accuracy of the theoretical calculations carried out. This has been due, as already explained, to the ambitious scope of this master's thesis, the delay in the reception of some fundamental parts of the robot or the unforeseen eventualities that a project of these characteristics always implicates. Notwithstanding this fact, that could be firstly mistaken for a result of a misled planning of the project, actually highlights one of its most remarkable facts. This is, that the achievement of the intended goals to their current extent has been carried out in a notable time lapse of 4 months.

11.2 The simulation and control architectures

The goal of a more general simulation environment than the existing LPZ robots used with the DACbot has been accomplished through the implementation of a Gazebo+ROS Control structure. The system stands out for facilitating the implementation of locomotion

controllers and the possibility of testing them in the simulation and in the real robot. Two have been created and tested in simulation, although not in the real setup. The simulation has proved to be a good tool for qualitative analysis but not for tests assuming real-world conditions, since that still needs further development.

11.3 The mathematical framework

The mathematical framework constructed has been utilized to simulate the dynamics of the robot and model the physics of a vertical jump on one leg. Its application has led the motor+gearbox selection process. Its use has also provided for a theoretical background to lead the design of the geometrical and inertial parameters and justify the criteria that has led the mechatronics design of the robot. Thus, the advantages of reducing the moments of the inertia of the limbs, rearranging their masses or finding the right proportion of their lengths has been provable and quantifiable thanks to this set of equations.

11.4 The test bench for experimentation

The devised test bench whose goal was the constraint of the robot movements to its sagittal plane has been constructed and installed on top of the existing treadmill. However, its functionality has not been tested further than manually because of the problems with the robot prototype described above.

Chapter 12

Conclusions and further work

As a general conclusion, the creation of a coherent simulation environment and the construction of a robust, low-cost and reconfigurable bipedal locomotion platform has been achieved. The ultimate goal was to contribute on the on-going line of research in locomotion control at the AI department at the Maersk Mc-Kinney Møller Institute, and RuBi offers a good compromise between features and affordability. From the bringing up of the robot to the set up of the simulation environment, including easy manufacturability and maintenance or code examples, RuBi is focused on facilitating the development of new locomotion controllers for teaching or research. The platform aims to create a bridge between the creation of controllers and their testing, which ultimately will speed up the development workflow and the possible experiments to be carried out in the area by the Embodied AI & Neurorobotics Lab.

However, much work was also planned to be left for further improvements of the RuBi study framework since, as said before, from the beginning this project was meant to be continued. This reason has led to the creation of the detailed and comprehensive documentation contained in this report, its appendixes, the assembly manuals¹ and the online repository [59]. Their aim is to provide the tools to easily and quickly understand and master the use of RuBi. From here, the uses of the platform are left to the imagination of future users.

12.1 Further work

The missing work that laid on the original scope of this thesis but could not be conducted is mentioned here for each of the main areas of the project. Besides, some enhancements and experimental setups already suggested are summarized below.

From the robot mechanics perspective, the very basic tests conducted seem to yield the need of reinforcing the hip structure, keeping its original design but widening its size. Also the robot holder on the external structure needs to be 3D-printed again to adjust its clearances due to the use of a different 3D printer for its creation, because of its size. The fastening of the motor interfaces could also be improved as it has shown to be a

¹<https://www.youtube.com/watch?v=rceASqlJ4HQ>

valid design but with some leaks when clamping. Other future improvements could be the development of a more accurate method to measure the tension of the belts or a redesign of the parallel spring holder to make faster the change of the torsion spring. Besides, in 8 it was suggested the construction of a small platform with additional contact sensors for experiments in vertical jumps. This could be necessary since the current contact sensors are placed on the heels, but the first landing point when hopping is the toes. Since a solution to the placement of more sensors on the current foot design could not be found, it is proposed to install off-board sensors to supply that information.

The calculations carried out to model the robot and its application have been created in MATLAB and used to evaluate the needs on hardware for RuBi. The models are left in an early development stage since they were meant to rely on experimental data to assess its accuracy and enhance them. However, with a fully operative prototype of the robot, the dynamic model can be further tested and used to detect and carry out improvement on the robot dynamics for a better performance, as it has been used for during its design. The construction of the dynamic model of a robot must be led by the seek of a balanced trade-off between accuracy in the representation of the actual device and simplicity to handle and understand its behavior. This stage has been tried to be found here through the use of the assumptions about the real world and the robot capabilities. A first suggestion for further improvement of the model would be to minimize these assumptions and try to model every dynamic term and scenario. However, as explained, this does not necessarily yield a better output of the process. Therefore it is left as further work the adaption of the mathematical model as required by the user in the need of a more advanced solution, since assessing the quality of a model is not a trivial task and depends on its use.

The electronic implementation of RuBi has proved to be robust and valid, but the set of possibilities that offers has not been fully exploited. The ROS Control interface has to be extended to include position control and readings from new sensors besides the angular encoders in the motors. The feet contact sensors have been installed but not plugged to the processor due to lack of documentation on the main board. This could be easily fixed adding some external circuitry as an Arduino, for instance, or exploring the capabilities of the existing I/O pins on the motor controller boards. Furthermore, abilitating new GPIOs on the hardware system could allow for more and more complex on-board sensors as an IMU or light sensors to improve self-awareness. Currently the relative encoders on the motors are manually initialized by placing the joints on their mechanical maximum extension. This should be enhanced to create an automatic solution.

Ultimately, the simulation model of RuBi was designed with direct drive transmission in the joints. The addition of springs to the model could lead to more realistic results, though this has to be tested. The adjustment of certain simulation values as the time step or some in the model of the robot, as the dynamic rotational friction, have to be done based on results taken from the real robot. The model can also be improved by attaching more realistic visual characteristics like adding colors and textures to the parts. The possibilities of Gazebo have not been squeezed so the inclusion of new simulation environments, sensors or other models is left to further research.

Bibliography

- [1] Sigmon B. A. *Bipedal behavior and the emergence of erect posture in man*. American Journal of Physical Anthropology, 2005.
- [2] Al-Shuka H. F. N. et al. "Dynamic Modeling of Biped Robot using Lagrangian and Recursive Newton–Euler Formulations". In: *International Journal of Computer Applications* (2014).
- [3] Cappellini G. et al. "ISO 7250-2". In: *Journal of Neurophysiology, American Physiological Society* (2006).
- [4] Cappellini G. et al. "Motor Patterns in Human Walking and Running". In: *Journal of Neurophysiology, American Physiological Society* (2006).
- [5] Cuevas E. V. et al. "Bipedal robot description". In: *Technical Report, Freie Universität Berlin, Institut für Informatik* (2005).
- [6] Di Canio G. et al. "Adaptive Combinatorial Neural Control for Robust Locomotion of a Biped Robot". In: *14th International Conference on the Simulation of Adaptive Behavior* (2016).
- [7] Grizzle J.W. et al. "MABEL, A New Robotic Bipedal Walker and Runner". In: *American Control Conference* (2009).
- [8] Hutter M. et al. "STARLETH: A Compliant Quadrupedal Robot For Fast, Efficient, and Versatile Locomotion". In: *International Conference on Climbing and Walking Robots (CLAWAR)* (2012).
- [9] Kale M. et al. "Relationships Among Jumping Performances and Spring Parameters During Maximum Speed Phase in Sprinters". In: (2009).
- [10] Kale M. et al. "The Relationship Between Running Speed and Measures of Vertical Jump in Professional Basketball Players: a Field-Test Approach". In: (2009).
- [11] Kojouri Manesh M. et al. "Simulation and Control of a Biped Walking Robot using Kinematic and Dynamic Modeling". In: () .
- [12] Noami K. et al. *Hydraulically Actuated Hexapod Robots*. Springer, 2014.
- [13] Pfeifer R. et al. "Embodied Artificial Intelligence: Trends and Challenges". In: *International Conference and Research Center for Computer Science* () .
- [14] Radkhah K. et al. "Detailed dynamics modeling of BioBiped's monoarticular and biarticular tendon–driven actuation system". In: *RSJ International Conference On Intelligent Robots and Systems* (2012).
- [15] Raibert M. et al. "Dynamic Legged Locomotion in Robots and Animals". In: *Progress Report to Office of Naval Research* (1995).

- [16] Sakka Sophie et al. "Humanoid Vertical Jumping based on Force Feedback and Inertial Forces Optimization". In: *International Conference on Robotics and Automation* (2005).
- [17] Satoshi Kagami et al. "Measurement and comparison of humanoid H7 walking with human being". In: *Robotics and Autonomous Systems* (2004).
- [18] Silvan M. F. et al. "A historical perspective of legged robots". In: *Journal of Vibration and Control* (2007).
- [19] Sánchez-Sánchez P et al. "Simplified Methodology for Obtaining the Dynamic Model of Robot Manipulators". In: *International Journal of Advanced Robotic Systems* (2012).
- [20] Tamada T. et al. "High-speed Bipedal Robot Running Using High-speed Visual Feedback". In: *14th IEEE-RAS International Conference on Humanoid Robots* (2014).
- [21] Tsagarakis N.G. et al. "An Asymmetric Compliant Antagonistic Joint Design for High Performance Mobility". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2013).
- [22] Vanderwerker E. E. et al. "A Brief Review of the History of Amputations and Prostheses". In: () .
- [23] Bullet. URL: <http://bulletphysics.org/wordpress/>.
- [24] *Calculation of power rating and drive centre distance*. Standard. Geneva, CH: International Organization for Standardization, Nov. 1987.
- [25] RepRap community. *ArcCompensation*. 2008. URL: <http://reprap.org/wiki/ArcCompensation> (visited on 07/19/2015).
- [26] *Cura slicer*. URL: <https://github.com/Ultimaker/Cura>.
- [27] Karssen J. G. D. "Robotic Bipedal Running". In: *Technische Universiteit Delft* (2013).
- [28] Dart. URL: <http://dartsim.github.io/>.
- [29] Lars-Peter Ellekilde and Jimmy A. Jorgensen. "RobWork: A Flexible Toolbox for Robotics Research and Education". In: *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)* (2010), pp. 1 –7.
- [30] Pscharakis S. G. "Dynamics of Vertical Jumps". In: *Edinburgh Napier University* ().
- [31] Walter W. G. "An Electro-Mechanical Animal". In: (1950).
- [32] *GAZEBO. Robot simulation made easy*. 2002. URL: <http://gazebosim.org/>.
- [33] T. et al. Geng. "A Reflexive Neural Network for Dynamic Biped Walking Control". In: *Neural Computation* (2006).
- [34] T. et al. Geng. "Fast Biped Walking with A Sensor-driven Neuronal Controller and Real-time Online Learning". In: *International Journal of Robotics Research* (2006).
- [35] *Go Robots*. URL: https://github.com/pmanoonpong/gorobots_edu.
- [36] Asada H. H. *Introduction to Robotics*. 2015.
- [37] Nicholas Herbert et al. "A preliminary investigation into the development of 3-D printing of prosthetic sockets". In: *Journal of rehabilitation research and development* 42.2 (2005), p. 141.

- [38] Klee. *KLEE-coil 302 000 030.160 M3 Hardened/zinc-plated yellow*. 2016. URL: <http://www.klee.dk/en/products/machine-, -din-norm-parts/kleecoils/self-cutting-types/kleecoils-type-302/12302000030160> (visited on 05/05/2016).
- [39] Johns Hopkins University Applied Physics Lab. "The Hopkins Beast". In: (1960).
- [40] Henry Louis Langhaar. *Dimensional analysis and theory of models*. Vol. 2. Wiley New York, 1951.
- [41] Jørgen Christian Larsen and Kasper Støy. "LocoKit-A Construction Kit for Exploration of Morphology of Legged Robots". In: (2011).
- [42] Locokit Electronics. URL: <http://www3.sdu.dk/cms/locokit/index.php?page=electronics>.
- [43] Grimmer M. "Powered Lower Limb Prostheses". In: *Technische Universität Darmstadt* (2015).
- [44] M Prime. URL: <https://mprime.io>.
- [45] M Prime One. URL: https://github.com/M-Prime/M_Prime_One.
- [46] F. et al. Martini. *Fundamentals of Anatomy and Physiology*. 5th Ed. Prentice Hall, 2001.
- [47] Maxon motors. *Maxon 14-225-en*. 2015. URL: http://www.maxonmotor.com/medias/sys_master/root/8816806527006/15-257-EN.pdf.
- [48] Maxon motors. *Maxon 14-264-265-en*. 2015. URL: http://www.maxonmotor.com/medias/sys_master/root/8816808460318/15-295-296-EN.pdf.
- [49] Lucas Nogueira. "Comparative Analysis Between Gazebo and V-REP Robotic Simulators". In: () .
- [50] ODE. URL: <http://www.ode.org/>.
- [51] OSG. URL: <http://trac.openscenegraph.org/projects/osg/>.
- [52] Overo Air Technical Specifications. 2015. URL: <http://media.gumstix.com/docs/gs3503a.pdf>.
- [53] Jerry E Pratt et al. "The RoboKnee: an exoskeleton for enhancing strength and endurance during walking". In: *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*. Vol. 3. IEEE. 2004, pp. 2430–2435.
- [54] *Prosthetics and orthotics – Vocabulary – Part 1: General terms for external limb prostheses and external orthoses*. Standard. Geneva, CH: International Organization for Standardization, Aug. 2011.
- [55] Dawkins R. "Why don't animals have wheels?" In: *The Sunday Times* (1996).
- [56] Research Network for Self-Organization of Robot Behavior. *LPZ Robots*. URL: <http://robot.informatik.uni-leipzig.de/software/>.
- [57] Coppelia Robotics. *v-rep*. URL: <http://www.coppeliarobotics.com/>.
- [58] *ROS Control*. URL: http://wiki.ros.org/ros_control.
- [59] *RuBi-Robot*. URL: <https://github.com/RuBi-Robot>.
- [60] Kondo Shiro. "Primate morphophysiology, locomotor analyses, and human bipedalism". In: *University of Tokyo Press* (1985).
- [61] Simbody. URL: <https://simtk.org/projects/simbody/>.

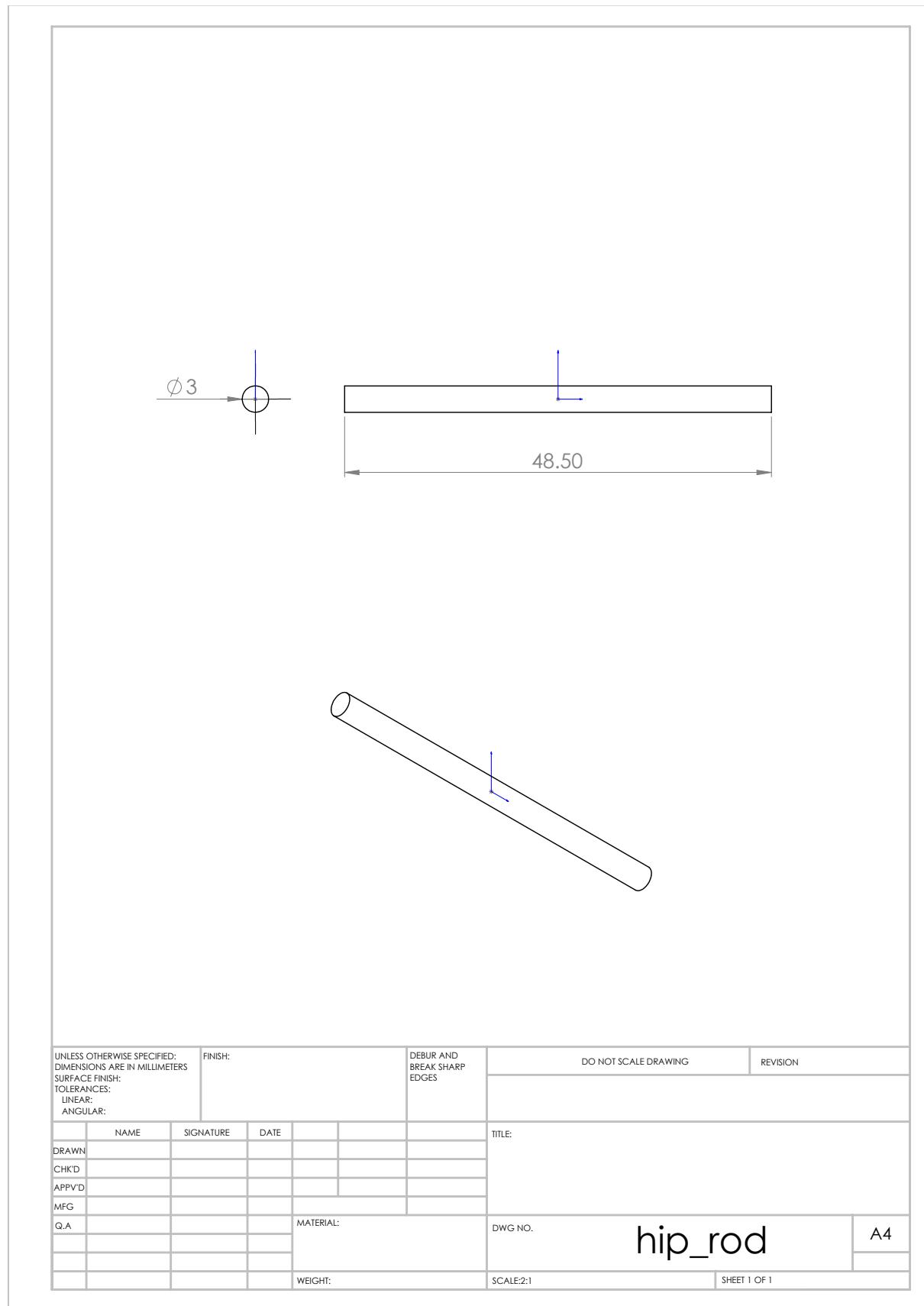
- [62] *Simplify 3D*. URL: <http://simplify3d.com/>.
- [63] Mona Singh. "U-SCRUM: An agile methodology for promoting usability". In: *Agile, 2008. AGILE'08. Conference*. IEEE. 2008, pp. 555–560.
- [64] *Slic3r*. URL: <http://slic3r.org/>.
- [65] K. et al. Sreenath. "A Compliant Hybrid Zero Dynamics Controller for Stable, Efficient and Fast Bipedal Walking on MABEL". In: () .
- [66] Aaron Staranowicz and Gian Luca Mariottini. "A survey and comparison of commercial and open-source robotic simulator software". In: *Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments*. ACM. 2011, p. 56.
- [67] John Hsu Steven Peters. *Comparison of Rigid Body Dynamic Simulators for Robotic Simulation in Gazebo*. 2014. URL: http://www.osrfoundation.org/wordpress2/wp-content/uploads/2015/04/roscon2014_scpeters.pdf.
- [68] *Synchronous belt drives – Metric pitch, curvilinear profile systems G, H, R and S, belts and pulleys*. Standard. Geneva, CH: International Organization for Standardization, Dec. 2014.
- [69] Dassault Systemes. *SolidWorks*. URL: <http://www.solidworks.com/>.
- [70] McGeer T. "Passive Dynamic Walking". In: *The International Journal Of Robotics Research* (1990).
- [71] *The Evolution of The Humanoid Robot*. URL: http://www.bibliotecapleyades.net/ciencia/ciencia_artificialhumans59.htm.
- [72] *The Robot Operating System*. 2007. URL: <http://www.ros.org/>.
- [73] *Trello*. URL: <https://trello.com/>.
- [74] N. G et al. Tsagarakis. "COMpliant huMANoid COMAN: Optimal joint stiffness tuning for modal frequency control". In: *Robotics and Automation (ICRA)* (2013).
- [75] *Tutorial: ROS Control*. URL: http://gazebosim.org/tutorials/?tut=ros_control.
- [76] Jan F Veneman et al. "Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation". In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 15.3 (2007), pp. 379–386.
- [77] Hurst J. W. *The Role and Implementation of Compliance in Legged Locomotion*. 2008.
- [78] Jürgens H. W. "Erhebung anthropometrischer Maße zur Aktualisierung der DIN 33 402 - Teil 2". In: *Schriftenreihe der Bundesanstalt für Arbeitsschutz und Arbeitsmedizin* (2004).
- [79] Adam B Zoss, Hami Kazerooni, and Andrew Chu. "Biomechanical design of the Berkeley lower extremity exoskeleton (BLEEX)". In: *Mechatronics, IEEE/ASME Transactions on* 11.2 (2006), pp. 128–138.

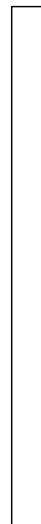
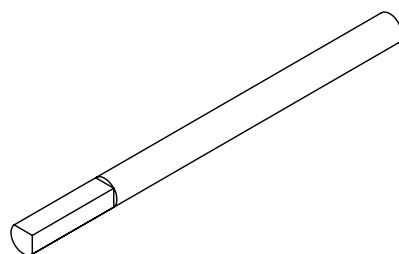
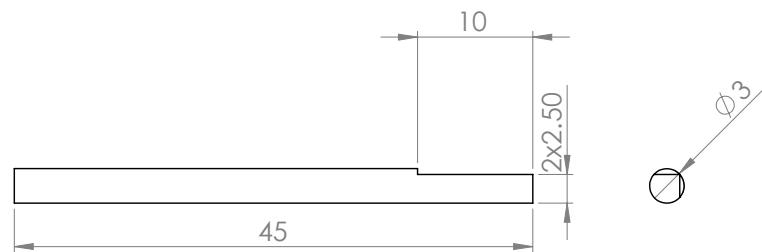
Appendices

.1 Profile selection

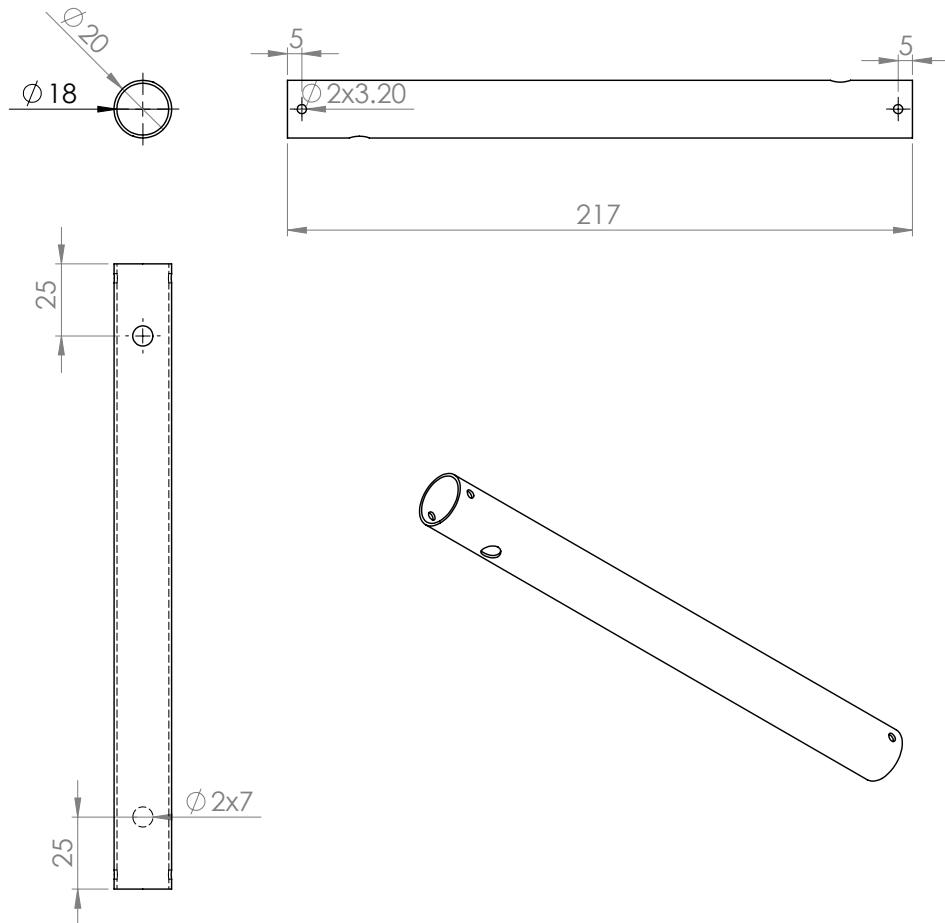
INPUT PARAMETERS					
Ultimate tensile stress [N/m] *	7.50E+08				
Maximum deformation allowed [mm]	5				
Elastic module [N/m] *	9.00E+10				
Density [kg/m^3]*	1.60E+03				
Total mass [kg]	1.50				
Jump height [m]	0.1				
Impact displacement [m]	0.005				
Knee Length [m]	0.2126				
Hip Length [m]	0.2622				
TORQUE IN KNEE [N*m]	62.57				
TORQUE IN HIP [N*m]	139.73				
FORCE IN KNEE'S EXTREME [N]	294.30				
FORCE IN HIP'S EXTREME [N]	294.30				
TUBES					
r1 [mm]	r2 [mm]	Intertia moment [m^4]	Stress knee [N/m^2]	Stress Hip [N/m^2]	Knee's displacement in extreme [mm]
0.005	0.006	5.27E-10	7.12E+08	1.59E+09	19.87
0.006	0.007	8.68E-10	5.05E+08	1.13E+09	12.07
0.007	0.008	1.33E-09	3.76E+08	8.40E+08	7.87
0.008	0.009	1.94E-09	2.91E+08	6.50E+08	5.41
0.009	0.01	2.70E-09	2.32E+08	5.17E+08	3.88
0.011	0.0125	7.68E-09	1.02E+08	2.28E+08	1.36
0.0125	0.015	5.44E-09	1.44E+08	3.21E+08	1.93
0.0135	0.015	1.37E-08	6.89E+07	1.53E+08	0.77
0.0185	0.02	3.37E-08	3.72E+07	8.30E+07	0.31
RODS					
r [mm]	Intertia moment [m^4]	Stress knee [N/m^2]	Stress Hip [N/m^2]	Knee's displacement in extreme [mm]	Hip's displacement in extreme [mm]
0.0015	3.98E-12	2.36E+10	5.27E+10	2634.28	2634.28
0.002	1.26E-11	9.96E+09	2.22E+10	833.50	833.50
0.0025	3.07E-11	5.10E+09	1.14E+10	341.40	341.40
0.003	6.36E-11	2.95E+09	6.59E+09	164.64	164.64
0.005	4.91E-10	6.37E+08	1.42E+09	21.34	21.34
0.006	1.02E-09	3.69E+08	8.24E+08	10.29	10.29
BOXES					
Interior side [mm]	Exterior side [mm]	Intertia side [mm]	Stress knee [N/m^2]	Stress Hip [N/m^2]	Knee's displacement in extreme [mm]
0.0025	0.004	1.81E-11	6.92E+09	1.55E+10	579.38
0.004	0.006	8.67E-11	2.17E+09	4.84E+09	120.85
0.007	0.008	1.41E-10	1.77E+09	3.96E+09	74.15
0.008	0.01	4.92E-10	6.36E+08	1.42E+09	21.29
0.017	0.02	6.37E-09	9.82E+07	2.19E+08	1.64
OPTIMAL KNEE'S PROFILE		Tube 20-18			3.32
OPTIMAL HIP'S PROFILE		Tube 20-18			4.09
					8.39
					6.29
					15.10
					46.57
					37.76

.2 Mechanical drawings

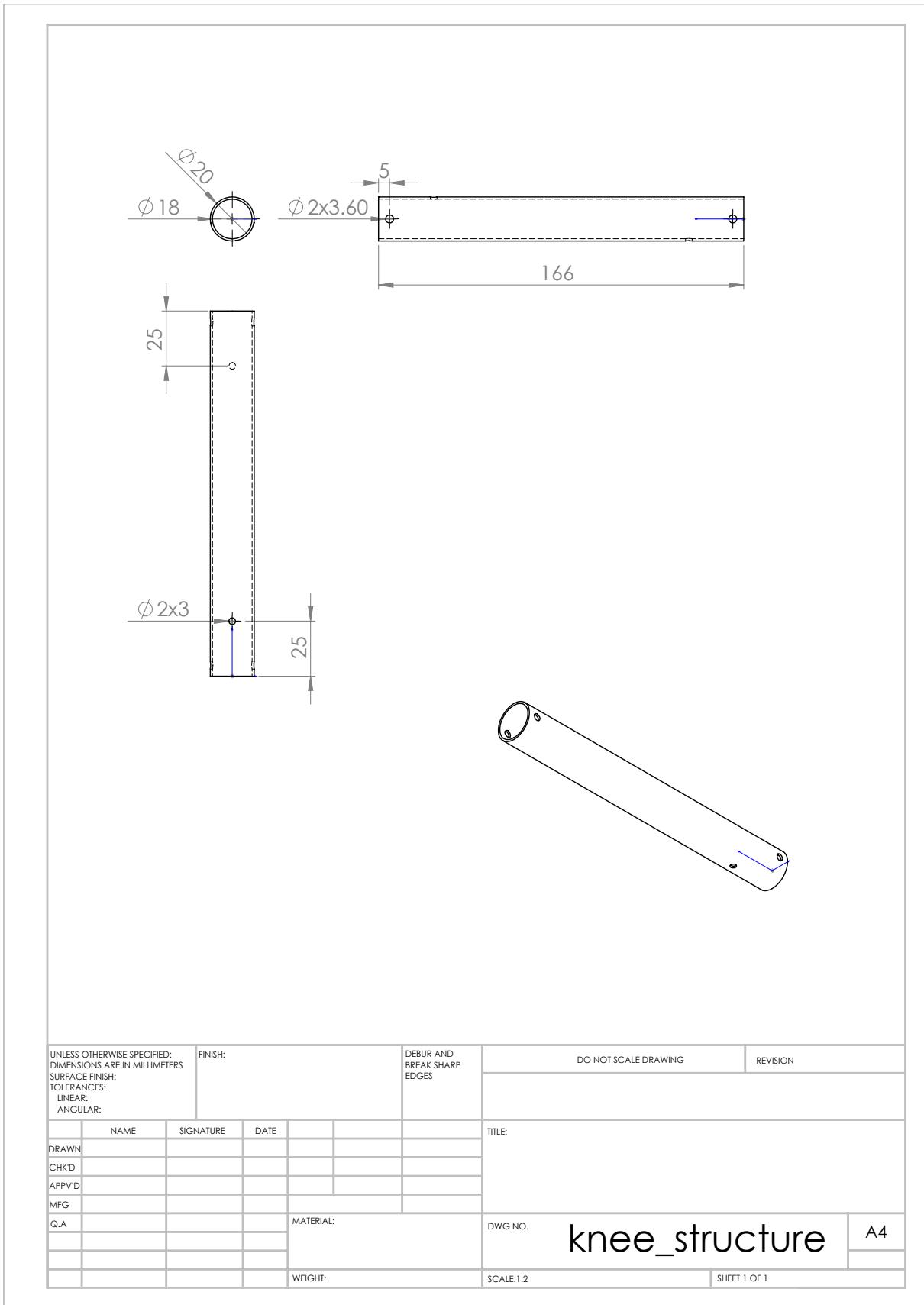


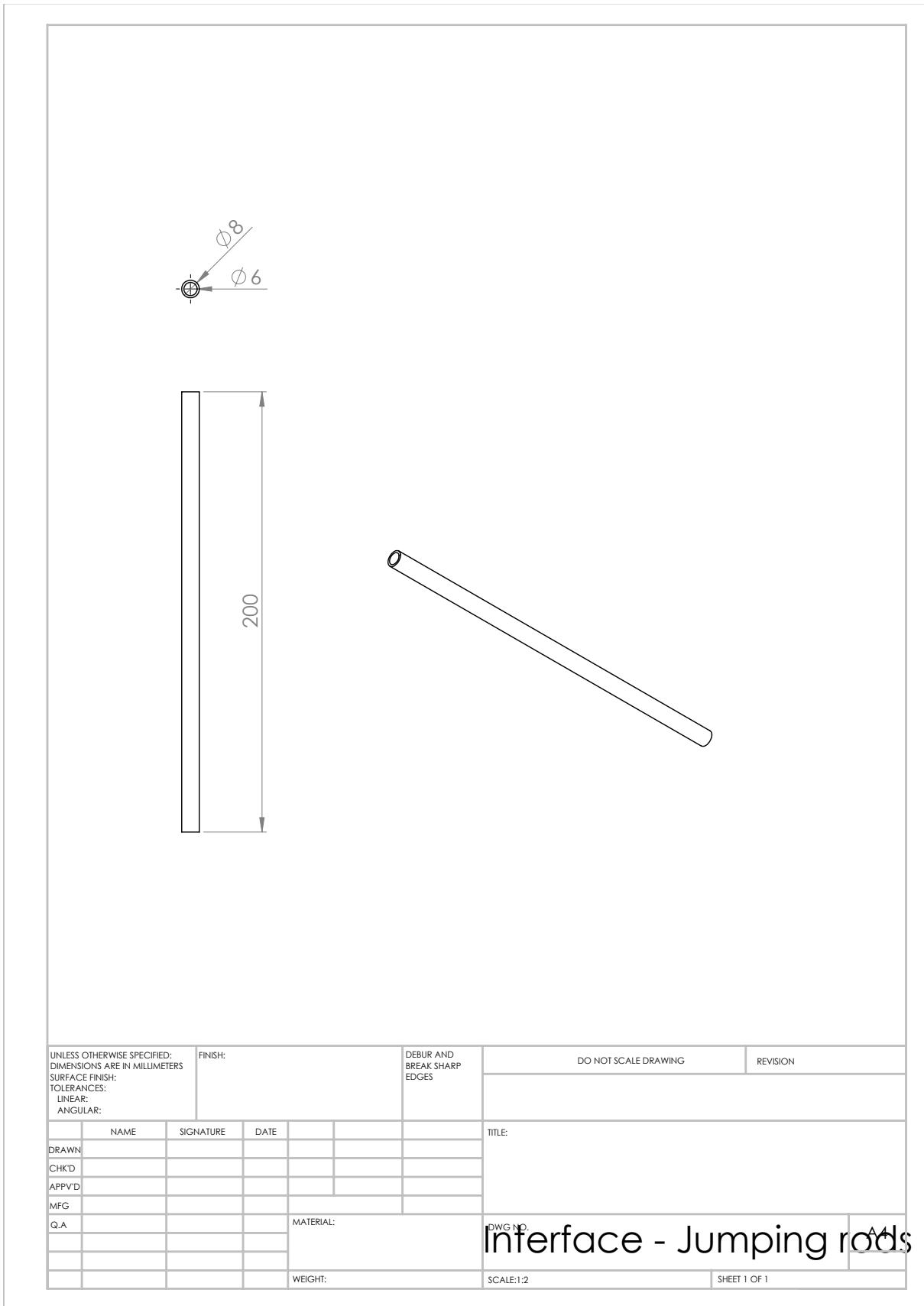


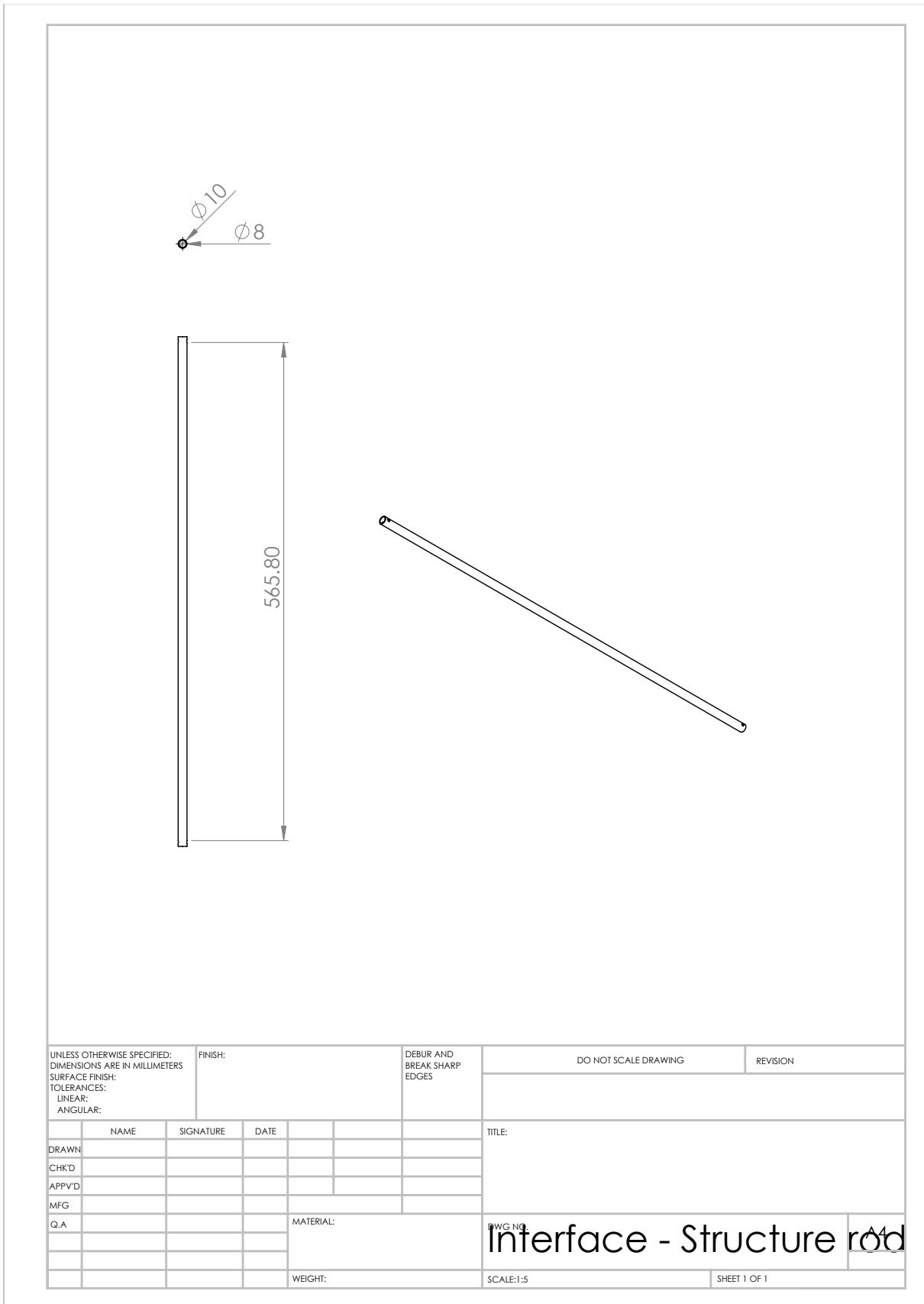
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBUR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING		REVISION
DRAWN							
CHK'D							
APP'D							
MFG							
Q.A				MATERIAL:	DWG NO.		
					knee and ankle rod	A4	
				WEIGHT:	SCALE:2:1	SHEET 1 OF 1	

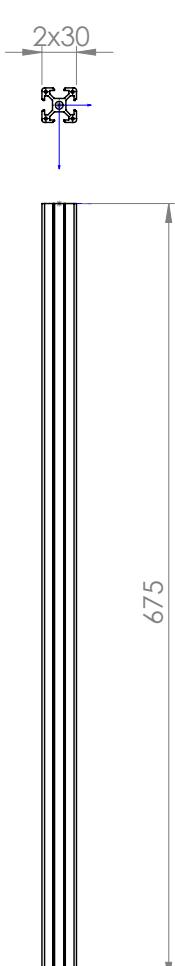


UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBUR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING	REVISION
DRAWN	NAME	SIGNATURE	DATE		TITLE:	
CHK'D					DWG NO.	
APP'D					hip_structure	
MFG					SCALE:1:2	
Q.A				MATERIAL:	SHEET 1 OF 1	
					A4	









UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBUR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING	REVISION
DRAWN	NAME	SIGNATURE	DATE	TITLE:		
CHK'D						
APP'D						
MFG						
Q.A			MATERIAL:	DWG NO.	Bosch Profile - 30x30 ⁴	
			WEIGHT:	SCALE:1:10	SHEET 1 OF 1	

.3 CAD parameters

```

"ankle_angle" = 70
"knee_angle"= 70
"hip_angle" = 30

"ankle_angle_max" = 120
"ankle_angle_min" = 50
"knee_angle_max" = 120
"knee_angle_min" = 0
"hip_angle_max" = 90
"hip_angle_min" = -60

"spring_ankle_parallel_arm_diameter"= 1.5
"spring_ankle_parallel_arm_length"= 13
"spring_ankle_parallel_intern_diameter"= 5.7
"spring_ankle_parallel_angle"= "ankle_angle"
"spring_ankle_parallel_turns"= 3
"spring_ankle_parallel_pitch"= "spring_ankle_parallel_arm_diameter" * 1.01
"spring_ankle_parallel_cutting_distance"= (1-"spring_ankle_parallel_angle" / 360) *
"spring_ankle_parallel_pitch"
"spring_ankle_parallel_nominal_diameter"= "spring_ankle_parallel_arm_diameter" +
"spring_ankle_parallel_intern_diameter"

"spring_ankle_serial_arm_diameter"= 1.91
"spring_ankle_serial_arm_length"= 10
"spring_ankle_serial_intern_diameter"= 5.7
"spring_ankle_serial_angle"= "90"
"spring_ankle_serial_turns"= 3
"spring_ankle_serial_pitch"= "spring_ankle_serial_arm_diameter" * 1.01
"spring_ankle_serial_cutting_distance"= (1-"spring_ankle_serial_angle" / 360) * "spring_ankle_serial_pitch"
"spring_ankle_serial_nominal_diameter"= "spring_ankle_serial_arm_diameter" +
"spring_ankle_serial_intern_diameter"

"spring_knee_parallel_arm_diameter"= 1.5
"spring_knee_parallel_arm_length"= 13
"spring_knee_parallel_intern_diameter"= 5.7
"spring_knee_parallel_angle"= -"knee_angle" + 90
"spring_knee_parallel_turns"= 3
"spring_knee_parallel_pitch"= "spring_knee_parallel_arm_diameter" * 1.01
"spring_knee_parallel_cutting_distance"= (1-( "spring_knee_parallel_angle") / 360) *
"spring_knee_parallel_pitch"
"spring_knee_parallel_nominal_diameter"= "spring_knee_parallel_arm_diameter" +
"spring_knee_parallel_intern_diameter"

"spring_knee_serial_arm_diameter"= 1.91
"spring_knee_serial_arm_length"= 10
"spring_knee_serial_intern_diameter"= 5.7
"spring_knee_serial_angle"= "90"
"spring_knee_serial_turns"= 3
"spring_knee_serial_pitch"= "spring_knee_serial_arm_diameter" * 1.01
"spring_knee_serial_cutting_distance"= (1-( "spring_knee_serial_angle") / 360) * "spring_knee_serial_pitch"
"spring_knee_serial_nominal_diameter"= "spring_knee_serial_arm_diameter" +
"spring_knee_serial_intern_diameter"

```

.4 Available springs

Table 1: Springs models (left configuration)

Springs available for experimentation	
Order number	Max torque (Nmm)
T032-090-172L	92,70
T035-090-187L	113,00
T038-090-234L	134,50
T040-090-187L	155,40
T045-090-203L	226,00
T048-090-218L	282,00
T051-090-234L	328,00
T054-090-296L	370,00
T059-090-296L	475,00
T063-090-343L	582,00
T070-090-359L	791,00
T075-090-375L	989,00
T078-090-406L	1102,00

.5 Gears of the hip

SYMBOL	VALUE	UNIT	TERM
	Coarse Pitch Involute 20 deg		Standard-Diametral Pitches
Pd	33.0000		Diametral Pitch
m	0.7697		Modular Pitch
ø	20.0000	deg	Pressure Angle
mg	2.0000		Ratio, 1:x
C	28.8636	mm	Center Distance
MA	1.9935	mm	Approach Length
MR	1.8310	mm	Recess Length
mp	1.6832		Contact Ratio

PINION			
Np	25.0000		Number of Teeth
Dp	19.2424	mm	Pitch Diameter
do	20.7818	mm	Major Diameter
dr	17.3182	mm	Minor Diameter
a	0.7697	mm	Addendum
b	0.9621	mm	Dedendum
	0.0000		Addendum Modification Coefficient
	0.0000	mm	Addendum Modification
db	18.0820	mm	Base Diameter
ht	1.7318	mm	Whole Depth
p	2.4181	mm	Circular Pitch
	0.2309	mm	Fillet Radius
B	0.0000	mm	Backlash
t	1.2090	mm	Tooth Thickness
F	1.0000	mm	Face Width
dw	1.3300	mm	Pin Diameter
M	21.0519	mm	Measurement Over Pins
	3.0000		Number of Teeth to Gage Over
	5.9501	mm	Chordal Measurement

GEAR			
Ng	50.0000		Number of Teeth
Dg	38.4848	mm	Pitch Diameter
Do	40.0242	mm	Major Diameter
Dr	36.5606	mm	Minor Diameter
a	0.7697	mm	Addendum
b	0.9621	mm	Dedendum
	0.0000		Addendum Modification Coefficient
	0.0000	mm	Addendum Modification
Db	36.1639	mm	Base Diameter
ht	1.7318	mm	Whole Depth
p	2.4181	mm	Circular Pitch
	0.2309	mm	Fillet Radius
B	0.0000	mm	Backlash
t	1.2090	mm	Tooth Thickness
F	1.0000	mm	Face Width
dw	1.3300	mm	Pin Diameter
M	40.3545	mm	Measurement Over Pins
	6.0000		Number of Teeth to Gage Over
	13.0364	mm	Chordal Measurement

