IVAN FISCHMAN EKMAN SIMÕES

**Biped gait controller with active perturbation
recovery**

São Paulo
2020

IVAN FISCHMAN EKMAN SIMÕES

**Biped gait controller with active perturbation
recovery**

Dissertation submitted to the Polytechnic
School of the University of São Paulo to
obtain the degree of Master of Science.

São Paulo
2020

IVAN FISCHMAN EKMAN SIMÕES

**Biped gait controller with active perturbation
recovery**

Corrected Version

(Original version is in the unit that hosts the Postgraduate Program)

Dissertation submitted to the Polytechnic
School of the University of São Paulo to
obtain the degree of Master of Science.

Concentration area:

3152 - Control and Mechatronics

Advisor:

Prof. Dr. Arturo Forner Cordero

São Paulo
2020

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 29 de Setembro de 2020

Assinatura do autor: Ivan Fischman

Assinatura do orientador: Arturo

Catálogo-na-publicação

Simões, Ivan

Biped gait controller with active perturbation recovery / I. Simões -- versão corr. -- São Paulo, 2020.

77 p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecânica.

1.Robótica 2.Controle (Teoria de Sistemas e Controle) 3.Sistemas Dinâmicos I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecânica II.t.

ACKNOWLEDGMENTS

I thank my family and friends, because without them this work would not be possible.

RESUMO

Os objetivos deste trabalho são desenvolver um controlador de marcha bípede e um algoritmo de recuperação de perturbações, ambos baseados na teorias de estabilidade *capture point* com o critério *N-step capturability*, validar os algoritmos através de simulações; e projetar um robô bípede que represente a marcha no plano sagital, capaz de servir como plataforma de testes para a pesquisa em locomoção com pernas.

A eficácia do controlador e o desempenho do algoritmo de prevenção de quedas são avaliados por simulações com o robô projetado, que impõe diferentes tipos de perturbações, como: empurrões, rampas e terrenos irregulares. A capacidade do controlador de modular a velocidade do robô também é demonstrada. O algoritmo de recuperação de quedas é abordado como uma solução para aumentar a robustez sem detrimento significativo a eficiência energética.

O controlador obteve sucesso nas simulações, demonstrando limitações (de velocidade) somente em terrenos descendentes. O algoritmo prevenção de quedas conseguiu aumentar significativamente a robustez de uma marcha padrão com boa eficiência energética.

Palavras-Chave – Robôs com Pernas, Robôs Bípedes, Capture Point, N-Step Capturability, Controlador de Marcha, Prevenção de Queda

ABSTRACT

The goals of this work are to develop a biped gait controller and an active fall recovery algorithm, both based on the capture point stability theory with the criterion of N-step capturability, validate this algorithms via simulations; and to design a biped robot that represents gait on the sagittal plane, capable of serving as a test-bed for legged locomotion research.

The effectiveness of the controller and fall recovery are assessed through simulations, with the designed robot, that imposes different types of perturbations, such as: pushes, ramps and irregular terrain. The controller ability to modulate the robot's velocity is also demonstrated. The fall recovery algorithm is addressed as a solution to increase robustness without significant detriment to energy efficiency.

The controller was proved successful in the simulations, it only demonstrated speed limitation on descending slopes. The fall recovery manages to significantly increase robustness of a standard, energy efficient gait.

Keywords – Legged Robots, Bipedal Robots, Capture Point, N-Step Capturability, Gait Controller, Fall Recovery

LIST OF FIGURES

Figure 1	Quadruped robots	15
Figure 2	Biped robots	16
Figure 3	Bipo - CAD	16
Figure 4	Bibliographic review	18
Figure 5	Sagittal plane	23
Figure 6	Five links model	25
Figure 7	Linear inverted pendulum - 2D	28
Figure 8	Control architecture	32
Figure 9	Desired trajectories - Nominal gait	35
Figure 10	Control architecture with active fall recovery	38
Figure 11	Nominal gait - Walking on a plane	40
Figure 12	Speed modulation during walk	41
Figure 13	Speed modulation, evaluation of forward and backwards velocities	42
Figure 14	Force magnitude over time for $Mag = 6$	43
Figure 15	Push forward applied at approx 1s, $Mag = 6$	44
Figure 16	Push backward applied at approx 1s, $Mag = -8.8$	44
Figure 17	Constant push backward applied at approx 1s, $-1.1 N$	45
Figure 18	Ramp, 40% incline	46
Figure 19	Ramp, -4% incline	47
Figure 20	Sinusoidal terrain $Amp = 0.01 m$, $Fre = 6$	48
Figure 21	Sinusoidal terrain $Amp = 0.005 m$, $Fre = 19$	48
Figure 22	Force magnitude over time for $Mag = 11.1$	49
Figure 23	Active fall recovery, trajectories - Simulation results	52

Figure 24	Active fall recovery, torques - Simulation results	52
Figure 25	Torques of the standard gait	53
Figure 26	Torques of the more robust gait	54
Figure 27	Modules	66
Figure 28	Bipo - Five links model	67
Figure 29	Bipo - Five links model, front and side view	68
Figure 30	Wheel as the point feet - Cinematic incompatibility 1	68
Figure 31	Wheel as the point feet - Cinematic incompatibility 2	69
Figure 32	Bipo - Preliminary seven links model	69
Figure 33	Finite elements analysis - Pulley	70
Figure 34	Boom assembly	70
Figure 35	Critical scenarios for motor sizing	73
Figure 36	Torque vs time for walking on a plane	75
Figure 37	Electronic diagram	76

LIST OF TABLES

Table 1	Robot dimensions	26
Table 2	Bézier points (b_o) for the torso angle trajectory	33
Table 3	Bézier points (b_o) for the CoM height trajectory	34
Table 4	Bézier points (b_o) for the swing foot y trajectory	34
Table 5	Bézier points (b_o) for the swing foot x trajectory	34
Table 6	Nominal reference values	36
Table 7	Initial reference values	50
Table 8	Modified CoM height r_{CoM}^{y0} - Simulation results	50
Table 9	Modified intended step duration t_{step} - Simulation results	50
Table 10	Modified x distance from swing foot to the CP R_c - Simulation results	51
Table 11	Fall recovery reference values	51
Table 12	Structural sizing, parts list - Bipo	67
Table 13	Structural sizing, parts list - Stabilization bar	71
Table 14	Torque to weight vs gear ratio and leg length	74

LIST OF ABBREVIATIONS

CoM	Center of mass
CoP	Center of pressure
CP	Capture point
CPG	Central pattern generator
CR	Capture region
DC	Direct current
DCM	Divergent component of motion
DoF	Degrees of freedom
FL	Feedback linearization
FR	Fall recovery
HZD	Hybrid zero dynamics
LIPM	Linear inverted pendulum model
MPC	Model predictive control
ROS	Robot operating system
USP	University of São Paulo
VHIP	Variable-height inverted pendulum
WHO	World health organization
ZMP	Zero moment point

LIST OF SYMBOLS

t	Time
g	Gravity
b	Viscous friction coefficient
T_{max}	Maximum actuation
m	Masses
J	Moment of inertia
l	Length of the link
$\circ$	Place holder for subscripts, superscripts or variables
l_{cm}^n	Distance from the nth joint to the nth CoM
mh	Point mass located at the hip
f_k	Force exerted on the ground by extensible leg of the 2D LIPM
z_0	Constant height for CoM of the 2D LIPM
E_{LIP}	Linear inverted pendulum orbital energy
$r_{CP}(t)$	Capture point position at the instant t
r_{ankle}	Position of the support foot
t_{end}	Time at the end of the step
d_N	Radius of the (N)-Step CR
L_{max}	Maximum length of a step
$Dr_{foot}^x(t_{end})$	Desired foot x position at the end of the step
t_{step}	Total intended duration for the step
R_n	Reference value n
B_n	Point n of a Bézier curve
S	State of the robot
β	Desired trajectories
T	Torques
γ	Torso angle

c	Desired distance from the CP to take the step
D_γ	Desired torso angle/angle trajectory
R_γ	Reference torso angle
Dr_{foot}^x	Desired foot x trajectory
r_{foot}^x	Swing foot x position
r_{foot}^y	Swing foot y position
r_{CoM}^y	Height for the robot's CoM
r_{CoM}^y0	Reference height for the robot's CoM
a	Reference height for the swing foot
$D_\circ$	Prefix for desired value or trajectory
$R_\circ$	Prefix for reference value
$r_{CP}^N(t_{step})$	CP position at the end of the step for the nominal gait
τ	Motor torque
m_m	Motor weight
p	Proportionality factor between motor weight and robot weight
l	Femur length
r	Gear ratio
τ^j	Joint torques
F_{per}	Horizontal force perturbation
$b_\circ$	Points for the Bézier curves
$B(t, b_{i..n})$	Bézier curve
u	Control output
E	Decoupling matrix
v	FL control input
$\circ^{FR}$	Superscript for fall recovery trajectories, variables, and others
$\lambda_\circ$	Control gains
$e_\circ$	Error between value and desired value
$d_\circ$	Desired values
T_e	Tracking error

CONTENTS

1	Introduction	14
1.1	Goals	15
1.2	Text Structure	17
2	Literature Review	18
3	Design Architecture	23
3.1	Model	23
3.1.1	Assumptions	24
3.1.1.1	Impact Model	24
3.1.2	Five Links Model	25
3.2	Control Architecture	27
3.2.1	Stability Criterion	28
3.2.2	Controller Based on N-Step Capturability for the Five Links Model	30
3.2.2.1	Control Equations and Parameters	32
3.2.3	Perturbation Detection and Active Fall Recovery	36
4	Results	39
4.1	Simulations - Without Active Fall-Recovery	40
4.1.1	Walking on Flat Ground	40
4.1.2	Speed Modulation	41
4.1.3	Push	42
4.1.3.1	Constant Push	43
4.1.4	Ramp	46
4.1.5	Sinusoidal Terrain	47

4.2	Simulations - Active Fall-Recovery	49
4.2.1	Simulation Conditions	49
4.2.2	Parameters Evaluation	49
4.2.3	Fall Recovery Implementation	51
4.2.4	Performance vs Energy	51
5	Discussion	55
5.1	Controller	55
5.1.1	Active Fall Recovery	57
5.2	Five Links Model & Robot	57
6	Conclusion	60
6.1	Future Work	60
	References	62
	Appendix A – Project Bipo	65
A.1	Mechanical Design	65
A.1.1	Stabilization Bar	70
A.1.2	Actuators	71
A.1.2.1	Motor Sizing	72
A.2	Electronics Design	76

1 INTRODUCTION

Robots can aid humans in various activities: the aging society can benefit from robot companions that are capable of helping in activities of daily living; robots can be used in hazardous situations, such as firefighting or working in radioactive environments. They could also be used to increase efficiency or reduce delivery costs, as Ford and Agility Robotics recently presented (Ford Motor Company; Agility Robotics, 2019). To effectively accomplish these tasks, robots can benefit from the mobility provided by legged locomotion.

At the time of writing, the world is suffering the COVID-19 pandemic. A viral disease that started to spread from China, and has reached almost all regions on the globe in just a few months, having registered over 5 million cases – source: Newspaper “O Estado de São Paulo”, (CONCONI et al., 2020), accessed at 05/21/2020.

One of the recommended measures to slow the spread of COVID-19 is social distancing – source: World Health Organization (WHO) (WHO, 2020), accessed at 05/21/2020 – however, in many states of Brazil people have difficulties to maintain social distance rates at the recommended values – source: Newspaper “O Estado de São Paulo” (DOLZAN, 2020) accessed at 05/21/2020) – One factor that hinders the ability of a society to restrict social contact are the essential businesses. For instance, drugstores, supermarkets, delivery companies and many others are necessary for a society to function, and cannot stop despite of the pandemic. This, not only represents a high personal risk, but also might spread the disease. In this respect, if current robots would have the versatility to assist on essential tasks, then they could help to effectively fight contagious diseases by reducing the number of people that need to expose themselves. In this context, legged robots may present some advantages over other types of mobile robots, eg. lower noise and higher autonomy than drones and more capable of traversing challenging terrains than wheeled robots.

Recently, legged robots technology has matured enough to be used in actual tasks, outside the research laboratories. E.g. Boston Dynamics has started to commercialize

its quadruped robot Spot Mini, a robot intended to be a generic legged platform for various tasks. ANYbotics has presented ANYmal C, also a multi purpose quadruped robot, capable of inspecting industrial sites; and the Italian Institute of Technology has presented HyQReal, a hydraulic robot.

Figure 1: Quadruped robots

(a) Spot Mini

(b) ANYmal C

Source: (a) <https://www.bostondynamics.com/spot> (visited in 05/08/2019)
 (b) <https://www.anybotics.com/anymal-legged-robot/> (visited in 06/25/2020)

In contrast to quadruped robots, biped robots are not as common. Even though we can find examples of biped robots such as Atlas from Boston Dynamics, Cassie from Agility Robotics and Asimo from Honda, they are usually used for research. The complexity of biped walking dynamics is one of the difficulties that must be solved before these robots can be usefully applied to replace several human tasks. Despite being more complex, biped and humanoid robots can easily adapt to an environment made for and by humans.

In this work, we aim at enhancing the walking capability of an anthropomorphic biped robot. In order to do so, we proposed a robot design (see Fig. 3) that will serve as a test bed for different control strategies, and a biped gait controller able to control walking over different surfaces and to reject several types of gait perturbations.

In the following subsections the objectives of this work and an overview of the text structure are presented.

1.1 Goals

The goals of this work are:

- Design of a biped robot, named Bipo. This robot represents gait models restricted to the sagittal plane (two-dimensional models). The robot must have adequate com-

Figure 2: Biped robots

(a) Atlas

(b) Asimo

Source: (a) <https://www.bostondynamics.com/atlas> (visited in 05/08/2019)
(b) <https://global.honda/innovation/robotics/ASIMO.html> (visited in 05/08/2019)

Figure 3: Bipo - CAD

putational capabilities and actuation to allow implementation of different control strategies, and serve as a test bed for further research. The complete requirements for the robot are given in appendix A.

- Develop a controller based on N-step capturability to control the gait of the proposed robot. This controller must be robust against different surfaces (e.g., slopes) and disturbances (e.g., pushes). The complete requirements for the controller are given in section 3.2.
- Develop a perturbation detection algorithm and an active fall recovery strategy to increase robustness and efficiency of the proposed gait controller. The complete development is presented in the subsection 3.2.3.
- Validate all proposed algorithms via numerical simulations, that are presented in chapter 4.

1.2 Text Structure

After the brief introduction and the goals presented in this chapter, a bibliographic review can be found in chapter 2. Chapter 3 introduces the necessary hypothesis and the model used to describe the biped gait, followed by the theoretical background, the control formulation and the development of the perturbation detection and fall recovery strategy; results are presented at chapter 4, and the discussion in chapter 5. Finally the conclusions and future works are presented in chapter 6. Appendix A describes the design of the biped robot Bipo.

2 LITERATURE REVIEW

We carried out a bibliographic review with the purpose of identifying the main research lines and the state of the art in the area of biped gait control. We searched for the most recent relevant publications in order to get an in-depth view of the field. The following databases were used in the review: Scopus, Google Scholar, *Portal de Periódicos Capes* and the *Portal de Busca Integrada da USP*.

Figure 4, summarizes and classifies the different lines of research on biped robot gait control. It is important to note that very often different strategies can be combined to achieve the desired performance and they would fit into more than one classification. Wright et al. have a similar classification in (WRIGHT; JORDANOV, 2015).

Figure 4: Bibliographic review

The first classification is between methods that seek to model the phenomenon of walking using physical models and those that, instead of describing gait mathematically, seek solutions through other techniques, such as bio-inspired: CPG and biological neural networks (MANOONPONG et al., 2007) or general Machine Learning techniques.

The Central Pattern Generator (CPG) is a neural structure present in certain animals, responsible for the generation of locomotor patterns that result in gait. The detailed structure of CPG is discussed in (DUYSENS; FORNER-CORDERO, 2018, 2019). A review on the subject can be found at (IJSPEERT, 2008).

Due to the recent advancement in Machine Learning techniques, we can see a nearly exponential growth in the number of publications in the area of biped walking using these techniques. The work developed at the University of Michigan by Xingye Da et al. (DA; HARTLEY; GRIZZLE, 2017) showed a Supervised Learning application to generate a robust control strategy for various terrain variations and force disturbances. Three different Learning techniques were applied: Neural Networks, Linear Interpolation, and Support Vector Machines. To train the controllers, a gear library was generated using Hybrid Zero Dynamic (HZD) and optimal control.

The work of Da et al. (DA; HARTLEY; GRIZZLE, 2017) is an example of the combination of different strategies to achieve a better outcome. The gaits generated by the HZD technique, although optimal, were not suitable for rough terrain and disturbances. Supervised Learning was used to generate a versatile gait based on different optimized gaits. The authors reported an increase in the robustness of walking, but the lack of a camera to better evaluate the different inclinations of the terrain caused falls. The inclusion of a camera has been suggested as possible future work.

Methods that describe gait through physical models can be subdivided into methods that focus either on passive or active systems. McGeer (MCGEER, 1990) presented the analysis of a simple passive device that can walk stably when placed on a downhill ramp. This paper explained the dynamics that govern gait so that it can be combined with an active energy source to produce efficient walking.

There are methods that focus on joint position control, using a variety of approaches to define the adequate joint trajectories ensuring that the robot does not fall. Among these approaches there is the Zero Moment Point (ZMP), one of the first strategies used for biped gait generation. The formal definition of ZMP can be found in the work presented by Vukobratović and Borovac (VUKOBRATOVIĆ; BOROVAC, 2004):

ZMP is defined as that point on the ground at which the net moment

of the inertial forces and the gravity forces has no component along the horizontal axes.

If the ZMP exists, i.e. the point whose moment component is zero on the horizontal axes is within the robot's support base, it can be stated that the robot's foot is in dynamic balance and is not rotating around its extremities.

This metric is widely used in conjunction with trajectory optimization strategies that, when followed by robots, perform the gait. The work done by Kajita et al. (KAJITA et al., 2003) uses the concept of ZMP, a simplified cart-table model and preview control to generate gait.

Weservelt et al. (WESTERVELT et al., 2007) detailed the concepts of Hybrid Zero Dynamics (HZD) and Virtual Constraints in the book *Feedback Control of Dynamic Bipedal Robot Locomotion* which compiled and extended results from various publications (WESTERVELT; GRIZZLE; KODITSCHKE, 2003; GRIZZLE; ABBA; PLESTAN, 2001; CHEVALLEREAU et al., 2003). With these concepts it is possible to prove analytically the existence of an asymptotically stable limit cycle for a class of biped robots.

Virtual Constrains is the name given for the robot coordination control, that control the robot joints as a function of other parameters such as ankle angle (or other monotonically growing parameter along of the gait path). This control creates a kinematic constraint between the parameters. When these constraints are respected, the robot dynamics are restricted, that is, if the robot has M degrees of freedom (GL) and N constraints, the dynamics (over the continuous phase) of the complete robot can be described by dynamics with $M - N$ GL, this being called the Zero Dynamics.

During the gait cycle, the continuous phases are separated by impacts. If Virtual Constrains continue to be respected even after the remapping of the velocities "during" the impact, the Zero Dynamics is invariant under the impact map. Thus, if the restrictions are respected during the continuous phase and considering the invariance under the instantaneous impact map, it is possible to describe the complete robot gait by a reduced model. The complete hybrid dynamics of the robot restricted to the Virtual Constrains results in the Hybrid Zero Dynamics.

The fact that the HZD has a small dimension (dimension 2 for each degree of sub-actuation) allows the complete stability analysis. The Poincaré map is used to prove the existence of a stable limit cycle for HZD and consequently for complete dynamics. Proving this existence analytically for a system with many degrees of freedom would not be feasible.

To find more efficient gaits, the functions that compose the Virtual Constraints are optimized subject to nonlinear constraints that ensure the gait is possible. These restrictions include those that guarantee the existence of a stable limit cycle in the system zero dynamics, no foot slipping and, for non-pointed feet, a restriction for the ZMP to ensure dynamic stability.

Kajita et al. (KAJITA et al., 2001) published in 2001 a paper presenting a simplified model to describe the walking of biped robots, the 3D inverted linear pendulum (3D LIPM). This model consists of a point mass supported by a massless leg of variable length. Restricting the height of the point mass to a plane parallel to the ground the model dynamics equations become linear and uncoupled.

A 2D version of the inverted linear pendulum model was used by Pratt et al. (PRATT et al., 2006) to introduce the concept of capture point (CP). Humans and humanoids that suffer large disturbances during gait need to take a step towards restoring balance. CP is the point on the ground where this step must be taken in order to reach the equilibrium position. The same metric was independently presented by both Hof et al. (HOF; GAZENDAM; SINKE, 2005) and Takenaka et al. (TAKENAKA; MATSUMOTO; YOSHIIKE, 2009) under the name of Extrapolated Center of Mass and Divergent Component of Motion (DCM) respectively.

Robert J. Griffin et al. (GRIFFIN et al., 2017) used a strategy based on tracking the error of the instantaneous capture point to modify the nominal ankle trajectory as well as the step time and length in order to increase the walking robustness of the humanoid robot Atlas. The algorithms allowed for real-time implementation. In addition, they presented a model predictive control (MPC) algorithm capable of combining both the ankle strategy, hip strategy and step adjustment with yaw adaptation to maintain balance Griffin e Leonessa (2019). The algorithms uses optimization and the time-varying DCM, allowing height control of the CoM. Simulations were conducted with ESCHER humanoid on Gazebo where performance was evaluate on different scenarios: walking on flat ground, terrain with varying heights and pushes.

The concept of CP was extended by Koolen et al. (KOOLEN et al., 2012) to the concept of N-step capturability for three simplified biped models: 3D LIPM, 3D LIPM with non-point foot and 3D LIPM with non-point foot and a reaction mass. For each model, the region on the ground where it is necessary to step, so that the robot reach the equilibrium position in up to “N” steps, is calculated.

Pratt et al. (PRATT et al., 2012) used a control algorithm based on N-step cap-

turability and the simplified 3D LIPM foot model, to control a 12-degree 3D robot. The model simplifications were needed to make the control calculations feasible. However, the errors caused by the differences between the robot and the model would result in one or two additional small adjustment steps for the robot to achieve equilibrium without compromising stability.

Also derived from the capture point concept, Rossi (ROSSI et al., 2019) developed a stability criterion: the Predicted Step Viability. This criterion uses the full dynamics of the simplified robot (such as the five link model Rabbit (CHEVALLEREAU et al., 2003)) to design the controller. The criterion determines the current step so that subsequent steps have the ability to bring the CP closer to the robot foothold, thus guaranteeing that the robot shall reach the equilibrium position in a finite number of steps. Optimizations and future step predictions are used to determine the gait. Although robust and versatile, it is a computationally demanding stability criterion, thus, posing some challenges for online implementation.

Koolen et al. (KOOLEN; POSA; TEDRAKE, 2016) aimed to shift away from models that limit the center of mass height. They developed a controller that reaches the equilibrium position by determining a CoM height trajectory, instead of other strategies of maintaining balance, eg.: controlling linear and angular momentum, shifting the center of pressure position, or taking a step. The controller was developed for a Variable-height inverted Pendulum (VHIP) model and was able to reach the balance for all the feasible initial conditions. Similarly, García (GARCIA; GRIFFIN; PRATT, 2019) has approached the problem of stability without CoM height requirements. In their work, they have analyzed the necessary and sufficient conditions for 0-Step capturability of a 3D VHIP and separated the 3D motion to implement feedback control laws to reach balance.

Each one of the methodologies presented has advantages and disadvantages, e.g., HZD considers the complete robot dynamics to synthesize the gait, which leads to accurate results on the actual implementation; however, the controller is specific for each robot, therefore it is not easy to use it in different robots. It also leads to a constrained gait, that cannot easily adapt to the terrain or perturbations. On the other hand, CP is based on a generic model and it is very robust against perturbations; however, it is sensitive to model errors, when the robot does not behave like the LIPM, the controller will have a poor performance. Step Viability allows optimal gait and versatility, but is not viable for online implementation. Therefore, this work aims to improve biped gait capabilities, detailed objectives are given in the next section.

3 DESIGN ARCHITECTURE

This section presents the theoretical background used to develop the dynamic model and the control architecture. The mechanical desing of the robot is presented on Appendix A. Although they are presented in a sequential order, their development has some overlap. The controller depends on the dynamic model, that in turn depends on the robot design; however, the controller also sets requirements for the robot. These dependencies must be taken in consideration during development.

3.1 Model

As a first step towards developing a gait controller, a simplified model has been used. The major simplification was to describe the gait dynamics only on the sagittal plane. Sagittal plane is the plane that divides the human body into two halves, right and left; see Fig. 5.

Figure 5: Sagittal plane

Source: <https://biologydictionary.net/sagittal-plane/> (visited in 09/08/2019)

This simplification results in a planar model instead of three-dimensional while still representing the main gait dynamics. This is a widely accepted simplification that has

been adopted by other authors in the literature (CHEVALLEREAU et al., 2003; PRATT et al., 2006; ROSSI, 2017; GOSWAMI; ESPIAU; KERAMANE, 1997).

3.1.1 Assumptions

Before formally presenting the models some assumptions must be stated:

1. The biped gait is considered as a hybrid dynamic system, composed by a sequence of continuous phases interspersed by instantaneous ones;
2. In the continuous phase, the robot is in single support, that is, one leg on the ground;
3. The instantaneous phase is when the robot changes the support leg (specific hypothesis for this phase are given in the subsection 3.1.1.1);
4. Both legs are identical, therefore all the continuous phases have the same dynamic equations, regardless of which leg is the support, left or right;
5. The upper body is represented by a single link, the torso;
6. The feet cannot “pull” the ground;
7. Contact forces must respect the friction coefficient between the foot and the ground;

If the robot has pointed feet, the contact with the ground is considered a rotational joint, and the robot cannot actuate directly on this degree of freedom; resulting in a degree of sub-actuation. If the robot has non-pointed feet, it can actuate the ankle joint and has no sub-actuation; but, as described by the ZMP theory (VUKOBRATOVIĆ; BOROVAC, 2004), if the actuation is too large, it may result in the foot rotating about its edge, adding a degree of sub-actuation.

3.1.1.1 Impact Model

The impact occurs when the swing leg hits the ground, and it is treated as completely rigid. It was implemented according to the formulation made by Westervelt et al. (WESTERVELT et al., 2007). The authors presented the following assumptions to formulate the mathematical equations:

1. “An impact results from the contact of the swing leg end with the ground;

2. The impact is instantaneous;
3. The impact results in no rebound and no slipping of the swing leg;
4. At the moment of impact, the stance leg lifts from the ground without interaction;
5. The externally applied forces during the impact can be represented by impulses;
6. The actuators cannot generate impulses and hence can be ignored during impact;
7. The impulsive forces may result in an instantaneous change in the robot's velocities, but there is no instantaneous change in the configuration;"

3.1.2 Five Links Model

The five links model, Fig. 6, represents a biped robot in the sagittal plane with knees and pointed feet. The model dimensions can be seen in table 1: b^n is the viscous friction coefficient on the nth joint; T_{max}^n is the maximum actuation on the joints; m and J are masses and moment of inertia respectively; l is the length of the link and l_{cm}^n is the distance from the nth joint to the nth CoM. The final goal is to control the designed robot, shown in Fig. 3; therefore the dimensions were obtained from its CAD model. The five links model is actuated in four joints, the exception being the ankle joint. Lagrange's Equations were used to obtain the dynamic equations.

Figure 6: Five links model

Table 1: Robot dimensions

Masses		
$m1, m5$	0.254	kg
$m2, m4$	0.78	kg
$m3$	3.861	kg
Momentum of inertia		
$J1, J5$	$4.3362 \cdot 10^{-4}$	$kg \cdot m^2$
$J2, J4$	$8.7409 \cdot 10^{-4}$	$kg \cdot m^2$
$J3$	$1.4148 \cdot 10^{-2}$	$kg \cdot m^2$
Viscous friction		
b	1	$N \cdot m \cdot s/rad$
length		
$l1, l2, l4, l5$	0.15	m
$l3$	0.2095	m
l_{cm}^1	0.090017	m
l_{cm}^2	0.084746	m
l_{cm}^3	0.071259	m
l_{cm}^4	0.065276	m
l_{cm}^5	0.059982	m
Joints Torque		
T_{max}	11.3	Nm

For the simulation, both the viscous coefficient and the maximum actuation for all the joints were set to be equal, since the robot has exactly the same motor/transmission in each joint. The viscous coefficient was an estimation based on the values of RABBIT (WESTERVELT et al., 2007), that had a higher gear reduction on its joints. The joint representing the point feet has no damping. This leads to a more accurate dynamic representation of the rotation of the robot around its feet; a fast dynamics that must be considered by the proposed controller.

3.2 Control Architecture

In order to perform different types of tasks, it is necessary that the biped robot is able to walk reliably. Hence, the controller must meet the following requirements:

- Robustness against perturbations – The controller must compensate perturbations either with feedback (use feedback to follow desired trajectories) or adapting the gait patterns (changing the foot placement trajectory, for example) to resist pushes or unexpected changes in the walking surface.
- Versatile – The controller must be able to ensure stability for a range of speeds, and adapt to the task and the terrain. For instance, a walking pattern generator that allowed gait adaptation to the terrain, proposed by Hong and Kim (HONG; KIM, 2013).
- Low energy consumption – In order to have a longer autonomy.
- Real-time implementation – The proposed controller is intended to control a real robot, therefore it must be feasible to run in real time, in order to meet this requirement the control loop should run at a rate of about 250 Hz (RAMOS et al., 2018).
- Generalizable – Possibility to apply it to different robots, without large modifications.

To achieve the desired performance for the robot Bipo, it was decided to base our control algorithm in the capture point (PRATT et al., 2006)/N-step capturability (KOOLEN et al., 2012) stability criterion. The same metric was independently presented by both Hof et al. (HOF; GAZENDAM; SINKE, 2005) and Takenaka et al. (TAKENAKA; MATSUMOTO; YOSHIIKE, 2009). This criterion is also similar to the multiple step strategy

to recover from trips by Forner-Cordero et al. (CORDERO; KOOPMAN; Van Der Helm, 2004).

In the following section a brief explanation about this criterion will be given.

3.2.1 Stability Criterion

As presented in chapter 2, the capture point can be informally defined as the point where the robot must step in order to reach a stop without falling (KOOLEN et al., 2012). In order to understand better the mathematical equations behind this theory a simple model such as the 2D linear inverted pendulum (2D LIPM) can be used, see Fig. 7. It consists of a point mass (m), and an extensible leg without mass or inertia that exerts a force (f_k) on the ground. It is assumed that the leg can be instantaneously placed in a desired position to perform a step.

Figure 7: Linear inverted pendulum - 2D

The mass is set to stay at a constant height z_0 , therefore f_k is defined as: $f_k = mg/\sin\theta$. The dynamics of the mass along the x axis can be described as Eq. 3.1; as presented by Prat et al. (PRATT et al., 2006).

$$\ddot{x} = \frac{g}{z_0}x \quad (3.1)$$

This is analogue to a mass-spring system, where the spring has a negative-rate stiffness. Because of this observation, the authors (PRATT et al., 2006) derive a conservative

quantity called “Linear Inverted Pendulum Orbital Energy” E_{LIP} :

$$E_{LIP} = \frac{1}{2}\dot{x}^2 - \frac{g}{2z_0}x^2 \quad (3.2)$$

It is now possible to find a displacement x for a given velocity that results in $E_{LIP} = 0$. If the Orbital Energy of the 2D LIPM is zero, the mass will decelerate until it reaches $x = 0$ and has no velocity, that is, the equilibrium position. Note that changing the displacement x means taking a step. The capture point is formally defined as:

$$r_{CP} = \dot{x} \sqrt{\frac{z_0}{g}} \quad (3.3)$$

Koolen et al. (KOOLEN et al., 2012) further explored the dynamics of the CP, and showed that during the continuous phase, the CP diverges from the support foot exponentially:

$$r_{CP}(t) = [r_{CP}(0) - r_{ankle}]e^t + r_{ankle} \quad (3.4)$$

$r_{CP}(t)$ is the position of the capture point at a given time t , and r_{ankle} is the position of the support foot. This equation is used to estimate the CP position at the end of the swing phase.

If it is not possible to reach $r_{CP}(t_{end})$ in the current step, that may not mean the robot will inevitably fall. The robot can perform several steps before finally reaching the CP. It can also be useful for walking to keep the robot stable without reaching the CP. Therefore, the authors (KOOLEN et al., 2012) presented the concept of the N-step capturability: If the robot steps into a N-step capture region (N-Step CR), it can reach a complete stop in at least N steps.

For the 3D LIPM with point feet, the N-Step CR are circles concentric with the CP. By definition, if the robot just stepped into a N-Step CR, it can now step into a (N-1)-Step CR, therefore:

$$r_{CP}(t_{end}) \leq d_{N-1} + L_{max} \quad (3.5)$$

d_{N-1} is the radius of the (N-1)-Step CR and L_{max} is the maximum length of a step. Combining equations 3.4 and 3.5 (for $r_{ankle} = 0$) gives: $r_{CP}(0) = (d_{N-1} + L_{max})e^{-t}$. At the end of the previous step, the distance between the swing foot and the instantaneous CP could not be more than d_N , hence $r_{CP}(0) = d_N$ and:

$$d_N = (d_{N-1} + L_{max})e^{-t} \quad (3.6)$$

Eq. 3.6 gives a recursive form to calculate all capture regions, where $d_0 = 0$. This series allow the computation of d_∞ in closed form:

$$d_\infty = d_0 + \sum_{N=0}^{\infty} [d_{N+1} - d_N] \quad (3.7)$$

$$d_\infty = L_{max} \frac{e^{-t}}{1 - e^{-t}} \quad (3.8)$$

3.2.2 Controller Based on N-Step Capturability for the Five Links Model

Based on the capture point theory, the goal of the control algorithm is to provide adequate trajectories to certain relevant variables, in such a way that the robot can walk without falling. In this context, the selected control variables are:

- Torso orientation (γ) – To perform useful tasks with the upper body while walking, the torso must remain within a certain inclination range. This range is determined by the task and the physical limitations of the robot.
- Vertical position of the swing foot (r_{foot}^y) – This is relevant to ensure ground and obstacle clearance. Note that this requirement may increase energy consumption.
- Center of Mass (CoM) height (r_{CoM}^y) – Imposing a vertical height for the CoM of the robot approximates the real robot of the Linear inverted pendulum, allowing a more accurate stability control. It is also necessary to perform useful tasks.
- Horizontal position of the swing foot (r_{foot}^x) – This is the most important parameter; as it was seen in the section (3.2.1), adequate foot placement is the key to guarantee stability.

The control architecture is divided into two main loops: Joint control loop, that is contained in the Stability control loop. The first one is responsible for controlling the trajectories of the joints using Feedback Linearization. The Stability control loop is responsible for defining the desired trajectories to guarantee stability. It consists of two parts: Stability controller and Motion planner. The Stability controller chooses the desired swing foot position at the end of the step based on the N-step capture regions, the state of the robot and a given reference value. It must respect physical limitations, such as maximum step length and minimum step time. The Motion planner receives the desired foot position and other reference values, then translates it into desired trajectories for all relevant variables, that are fed into the Feedback Linearization loop.

In the present work, the control algorithm was implemented in simulations; thus, both loops were executed at every simulation iteration. For the future implementation of this control algorithm in the physical robot, it is possible to define different operation frequencies for each control loop to minimize computation requirements.

The desired trajectories for the control variables are described as Bézier curves, as in (WESTERVELT et al., 2007). These curves are composed by eight points parameterized by the time. At the beginning of every step, the desired positions and velocities are set to be the actual positions and velocities, this reduces energy consumption and produces smoother movements.

The Stability control loop receives information from higher control levels, such as path and task planning. The higher control levels are not in the scope of this work, but their interaction is represented as the reference values mentioned above. Those are:

- R_1 – Desired step duration t_{step} ;
- R_2 – Reference angle for the torso orientation R_γ ;
- R_3 – Reference distance between the swing foot and capture point at the end of the step R_c ;
- R_4 – Reference height for the swing foot R_a .
- R_5 – Reference height for the robot's CoM r_{CoM}^{y0} .

These values can be directly modified to adapt the gait strategy, for example: lower swing foot height can decrease energy consumption but also decrease obstacle clearance.

Similarly, a lower CoM will result in higher energy consumption but larger maximum step length.

A block diagram of the control algorithm can be found in Fig. 8.

Figure 8: Control architecture

3.2.2.1 Control Equations and Parameters

As presented previously, the control algorithm relies on several parameters. These parameters shape the gait pattern and must be determined adequately to ensure stable walking. Their values will influence robustness, energy consumption and any other characteristic of the gait. Herein the values for the standard gait (nominal gait) will be presented. The control output (u) is the output of the decoupling control law of the Feedback Linearization (FL) (SLOTINE; LI, 1991):

$$u = E^{-1} \cdot (v - J) \quad (3.9)$$

where E and J are matrices obtained from the FL process. E is the decoupling Matrix. v is a control input that actuates directly on the linearized and decoupled control variables.

v is set as a PI controller (SLOTINE; LI, 1991). In this way, the control algorithm will behave as an impedance controller:

$$v = \begin{bmatrix} \ddot{d}_1 - 2\lambda_1(\dot{e}_1) - \lambda_1^2 e_1 \\ \ddot{d}_2 - 2\lambda_2(\dot{e}_2) - \lambda_2^2 e_2 \\ \ddot{d}_3 - 2\lambda_3(\dot{e}_3) - \lambda_3^2 e_3 \\ \ddot{d}_4 - 2\lambda_4(\dot{e}_4) - \lambda_4^2 e_4 \end{bmatrix} \quad (3.10)$$

λ are the control gains. If gains are set are too high, the robot may slip or “pull” from the ground. If they are too low, the robot will not follow the desired trajectories, thus, gait stability cannot be guaranteed. For the nominal gait, $\lambda = 150$; this value was manually tuned during the simulations. The desired values d_o and errors e_o are obtained from the state of the robot and the Bézier curves, $B(t, b_{i..n})$, that are a parameterized function of points b_i (WESTERVELT et al., 2007):

$$B(t, b_{i..n}) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i b_i \quad (3.11)$$

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}$$

The points are a function of the reference values and robot state $b_o(R_o, y)$. Particularly, the points for the horizontal position of the swing foot depend on the predicted position of the CP at the end of the step. The Bézier points used by the controller are calculated by a code similar to the pseudo-code presented in the tables 2, 3 4 and 5.

Table 2: Bézier points (b_o) for the torso angle trajectory

Input:	State of the Robot (y) Bézier Parameter (Par) R_2
1:	b_1 =Initial Angle;
2:	$b_2=b_1$ +Initial Velocity $\cdot 1/7 \cdot t_{step}$;
3:	$b_3=b_2+(R_2-b_2)1/4$;
4:	$b_4=b_2+(R_2-b_2)2/4$;
5:	$b_5=b_2+(R_2-b_2)3/4$;
6:	$b_6=R_2$;
7:	$b_7=R_2$;
8:	$b_8=R_2$;
Output:	Bézier Points

Table 3: Bézier points (b_o) for the CoM height trajectory

Input:	State of the Robot (y) Bézier Parameter (Par) R_5
1:	b_1 =Initial Height;
2:	$b_2=b_1$ +Initial Velocity $\cdot 1/7 \cdot t_{step}$;
3:	$b_3=b_2+(R_5-b_2)1/4$;
4:	$b_4=b_2+(R_5-b_2)2/4$;
5:	$b_5=b_2+(R_5-b_2)3/4$;
6:	$b_6=R_5$;
7:	$b_7=R_5$;
8:	$b_8=R_5$;
Output:	Bézier Points

Table 4: Bézier points (b_o) for the swing foot y trajectory

Input:	State of the Robot (y) Bézier Parameter (Par) R_4
1:	b_1 =Initial Angle;
2:	$b_2=b_1$ +Initial Velocity $\cdot 1/7 \cdot t_{step}$;
3:	$b_3=R_4$;
4:	$b_4=R_4$;
5:	$b_5=R_4$;
6:	$b_6=R_4$;
7:	$b_7=R_4$;
8:	$b_8=-R_4/2$;
Output:	Bézier Points

Table 5: Bézier points (b_o) for the swing foot x trajectory

Input:	State of the Robot (y) Bézier Parameter (Par) R_3, R_1
1:	$r_{CP}(t_{step}) = [r_{CP}(0) - r_{ankle}]e^{(1-Par)\cdot t_{step}} + r_{ankle}$
2:	$DF = r_{CP}(t_{step}) - R_3$;
1:	b_1 =Initial Position;
2:	$b_2=b_1$ +Initial Velocity $\cdot 1/7 \cdot t_{step}$;
3:	$b_3=b_2+(DF-b_2)3/5$;
4:	$b_4=b_2+(DF-b_2)4/5$;
5:	$b_5=DF$;
6:	$b_6=DF$;
7:	$b_7=DF$;
8:	$b_8=DF$;
Output:	Bézier Points

Both the torso angle and the CoM height have a fixed desired height provided by higher control levels (reference values). Therefore their Bézier curves are approximately a horizontal line. A small deviation can be noticed at the beginning of the step. This occurs to accommodate the changes in velocity occasioned by the impact, and initial positions/velocities different from the desired value, in a more energetically efficient way. Adjusting the beginning of the desired trajectory is also used on the other controlled variables.

The swing foot trajectories should allow for ground and obstacle clearance. To do so, it is possible to define the vertical foot trajectory as an arch, the arch height will determine the maximum obstacle clearance, at the cost of higher energy consumption. The horizontal displacement of the swing foot ends at the desired distance from the CP at the end of the step. To ensure correct foot placement and stability, the prediction of the CP position at the end of the step and the swing foot horizontal desired trajectory (Dr_{Foot}^x) are updated at every control loop cycle. The three trajectories: ($D\gamma$, Dr_{CoM}^y and Dr_{Foot}^y) are updated only once per step. An example of all the desired trajectories is presented in Fig. 9. The desired absolute angle is measured from the vertical axis.

Figure 9: Desired trajectories - Nominal gait

The reference values for the CoM and swing foot height were chosen according to the robot geometry; The reference angle for the torso orientation was chosen to maximize compatibility with tasks. The total prescribed step duration was based on human gait

and adjusted to the robot size.

To determine R_c it must be considered that the step must occur inside the Capture Region, thus any value that respects this condition is valid. Moreover, if R_c is kept constant, the gait will converge to a nominal step/gait. This can be inferred by evaluating Eq. 3.4 for a fixed step duration t_{step} :

$$r_{CP}(t_{step}) = [r_{CP}(0) - r_{ankle}]e^{t_{step}} + r_{ankle} \quad (3.12)$$

A fixed time step and a constant initial distance between the ankle and the Capture Point results in a constant distance between the ankle and capture point at the end of the step, for every step. For the LIPM, the convergence to a nominal step will occur instantaneously. However, for a more complex model, this convergence might take a few steps. This is a direct consequence of the differences between the LIPM and the robot model. The swing forward of the legs (with mass) influences the CP dynamics, accelerating it. This behavior and the impact dynamics are not predicted by Eq. 3.12; and they slow the convergence rate.

All values have also been adjusted by evaluating performance during simulations. The final values for the nominal gait are listed on table 6. The effects of changing R_c will be further explored in the next section and in section 4.1.2.

Table 6: Nominal reference values

Reference		Value	Unit
Intended step duration	t_{step}	0.4	s
Absolute torso angle	R_γ	0	rad
CoM height	r_{CoM}^{y0}	0.29	m
Swing foot height	R_a	0.01	m
x Distance from swing foot to the CP	R_c	0.002	m

3.2.3 Perturbation Detection and Active Fall Recovery

Since the trajectories for the swing foot are a function of the capture point, the stability control loop has already a builtin fall recovery, because the step is automatically adjusted in the direction of the disturbance. Nevertheless, the robustness against perturbations can be improved with larger changes to all the desired trajectories.

To choose the correct strategy, it is necessary to understand the intended results. The

stability criterion indicates that the robot will fall if the capture point is not reachable. Therefore, the trajectory changes must have the effect of preventing the capture point of getting further away (or even bringing it closer) to the robot stance foot.

By analyzing both equations 3.3 and 3.12 the key aspects can be identified that may be used to increase robustness:

- Decreasing R_c – From the evolution of the capture point over time, it can be observed that if the initial distance from the capture point to the stance foot is smaller, the capture point will diverge less (in the same amount of time).
- Decreasing t_{step} – Taking a step in less time will result in less time for the capture point to evolve naturally and further away from the robot.
- Decreasing r_{CoM}^{y0} – From the definition of the capture point, it can be seen that the CP position is proportional to the height of the CoM; therefore, if the CoM is lowered the CP will be brought closer to the robot.

If these modifications are not correctly tuned, they may have a negative effect. Faster steps may result in slipping, decreasing R_c can result in larger steps, that may not be viable and changing the height of the CoM can reduce the vertical ground reaction force and the foot may also slip. Moreover, these changes can influence heavily the CP dynamics in such a way that the simplified LIPM model is not valid.

Changes to these parameters may have other drawbacks, such as energy consumption/efficiency. Thus it is interesting to “activate” these strategies only when necessary. The final solution was to implement a “Perturbation Detection”, allowing the control to switch to the more robust trajectories only if a perturbation is identified.

The control algorithm is able to detect perturbations by monitoring the tracking error T_e (see equation 3.13) of the trajectories. If the error reaches a threshold, it indicates a perturbation. This threshold can be adjusted to make the algorithm more or less sensitive to the error. The perturbation detection has been limited to the first 0.1 s of the step in order to ensure that the trajectories don't change too abruptly after the perturbation detection. Once a perturbation is detected, the desired trajectories are updated the same way as in the normal gait controller with new values for the three variables presented above: R_c (R_3^{FR}), t_{step} (R_1^{FR}) and r_{CoM}^{y0} (R_2^{FR}). The numerical values of these new variables will be presented at section 4.2.

$$T_e = \sqrt{e_1^2 + e_2^2 + e_3^2 + e_4^2} \quad (3.13)$$

After the perturbation is detected the more robust trajectories are maintained for a few steps. This ensures a faster and more reliable convergence to the nominal gait. It also prevents a sequential activation and deactivation of the fall recovery strategy on the steps following the perturbation.

In Fig. 10 it can be observed an extended block diagram of the control algorithm with the inclusion of the active fall recovery module.

Figure 10: Control architecture with active fall recovery

4 RESULTS

The simulations presented here were conducted to assess the effectiveness of both the gait controller and the active fall recovery. During the simulations the robot was subjected to different disturbances with increasing magnitude until the limits of recovery were found. The disturbances applied were:

- Push at the middle or beginning of the step with limited duration
- Constant Push
- Slopes (positive and negative)
- Sinusoidal Terrain

Simulation including walking on flat ground and speed modulation on flat ground are also presented. The simulations were validated to ensure that they were representative of real situations. On previous simulations we found that it was not necessary to simulate for a large number of steps, since on the presented results the convergence to the nominal gait can already be observed. The simulations were checked for:

- Slip during the continuous phase.
- Slip at the impact.
- Negative reaction force from the ground during the continuous phase (the ground contact is modeled as a rotational joint, the mathematical equations allow the floor to “pull” the foot, which is not possible).
- Negative reaction force from the ground at the impact.
- Swing foot positive velocity in the normal direction of the ground to ensure ground clearance at the beginning of the step.

The numerical integrator *ode45* on Matlab was used to perform the calculations.

4.1 Simulations - Without Active Fall-Recovery

4.1.1 Walking on Flat Ground

Simulating the proposed controller on a horizontal flat ground results in a stable gait. The gait rapidly converges to normal repeatable cycle that we consider to be a limit cycle; it takes about three to four steps to do so, despite the initial conditions being relatively far from it. The simulation was initialized with the robot close to the equilibrium position, in single support:

$$y_0 = [\frac{\pi}{2} - \frac{\pi}{10}; \frac{\pi}{5}; -\frac{\pi}{10}; \pi + \frac{\pi}{10}; -\frac{\pi}{4.5}; 0; 0; 0; 0; 0]; \quad (4.1)$$

In Fig. 11 the simulation results are summarized in one graph. The four directly controlled variables are presented: Torso orientation γ (scaled), Vertical position of the swing foot r_{foot}^y , Center of Mass (CoM) height r_{CoM}^y , Horizontal position of the swing foot r_{foot}^x ; The desired trajectories are displayed with the same color as the variable, but with a dashed line. Alongside to those variables, the prediction of the C.P. at the end of the step $r_{CP}(t_{step})$ and the Horizontal displacement of the Center of Mass r_{CoM}^x are presented. All of the following simulations present similar graphs.

Figure 11: Nominal gait - Walking on a plane; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

4.1.2 Speed Modulation

Eq. 3.12 brings a important information: for a fixed step time, increasing the distance from the swing foot to the CP before the impact will increase the CP position at the end of the next step. This means that the further away from the CP the robot steps, the fastest it will walk. Another important thing to note is that if the robot steps in front of the CP, $r_{CP}(0) - r_{ankle} < 0$, the CP position at the end of the step would be smaller than the position of the CP at the beginning of the step, $r_{CP}(t_{step}) < r_{CP}(0)$, thus the robot would walk backwards.

To assess this behavior, 24 steps were simulated, starting at the nominal gait presented before, but varying the value for the desired distance between the CP and the swing foot at the end of the step. Starting at $R_c = 0.002$, it varied to $R_c = 0.004$, and then to $R_c = -0.004$. Simulation results can be seen in Fig. 12.

Figure 12: Speed modulation during walk; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

It can be observed that around 5 s there was a change in the gait from forward walking to backward walking. The simulated robot is capable of reaching a maximum forward speed of 0.4274 m/s and a backward speed of -0.1084 m/s. Even though the absolute value of R_c for walking forward is similar than for walking backward, the robot reaches a higher forward velocity. This occurs because of the error in the calculation of the capture point. The calculated C.P. value has a biased error due to the discrepancies between the

LIPM and the five links model. The actual capture point for the five links model was estimated, via simulation, to be about 0.0022 m behind the C.P. calculated for the LIPM ($R_c = -0.00165\text{ m}$). This value was obtained by searching for which value of c resulted in the “stepping in place” behaviour.

If consider the estimated C.P. position for the five links model and compare similar absolute values for R_c we can obtain very close results; see Fig. 13. Varying 0.001 m from the adjusted C.P. resulted in forward and backward velocities of 0.0511 m/s and -0.0466 m/s respectively. These results agree partially with what was expected as there are still some difference in the absolute speed. This suggests that the robot configuration (torso inclination and knees orientation) and trajectories change how much swinging the leg modifies the CP dynamics, causing different tracking errors and consequently different c values. Note that for this simulation the torso was kept straight, if this was not the case, the robot configuration could have more impact in the tracking errors.

Figure 13: Speed modulation, evaluation of forward and backwards velocities - For the first half of the simulation $R_c = -0.00265\text{ m}$, then $R_c = -0.00065\text{ m}$; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

4.1.3 Push

To simulate a push, while the robot is walking with the nominal gait presented in section 4.1.1, a horizontal force (F_{per}) is applied to the robot hip at the middle of the

third step. The force magnitude is described by Eq. 4.2. Mag is an input value and corresponds to the maximum magnitude the force will reach. The force magnitude over time can be seen in Fig. 14.

$$F_{per} = Mag * (sigmf(t, [100, 0.9]) - sigmf(t, [100, 1.1])) \quad (4.2)$$

Figure 14: Force magnitude over time for $Mag = 6$

During the simulations, Mag was increased until the robot was not able to resist the perturbation and fell. The perturbation was applied in both directions, forward and backwards. From the simulations, the maximum magnitudes that the robot can resist without falling are $Mag = 6 \text{ N}$ forward and $Mag = -8.8 \text{ N}$ backwards. Simulation for both cases can be seen in Figs. 15 and 16. Even though the perturbation was applied only during the second step, the controller was able to return to the nominal gait in a few steps.

4.1.3.1 Constant Push

Constant forces were also tested and the robot was able to resist a constant 0.4 N push forward and -1.1 N backwards, without any modification of the reference values or gait trajectories. Applying a constant horizontal force on the robot, heavily influences the

Figure 15: Push forward applied at approx 1s, $Mag = 6$; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

Figure 16: Push backward applied at approx 1s, $Mag = -8.8$; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

C.P. evolution over time. Since the proposed controller tries to “follow” the capture point, it is able to adapt to this disturbance by adequately placing the swing foot, behaving as a “built in” fall recovery.

In the backward constant push simulations, it can be seen that the robot walks backwards (see Fig. 17), even though the parameters of the nominal gait were set for walking forward. This is an excellent result, because it shows that the gait can adapt to perturbations in order not to fall.

Moreover, the same strategy used to modulate speed can be applied to overcome the horizontal force perturbations. By increasing the reference distance from the C.P. at the end of the step R_c , the robot can walk forward despite the negative force. This strategy can be used, not only to overcome perturbations, but to easily interact with the environment, for example when pushing boxes or carts.

Figure 17: Constant push backward applied at approx 1s, -1.1 N , The simulation initial state is the same that for the nominal gait limit cycle. It can be observed that around 1s the robot starts walking backwards due to the perturbation; then the R_c value is adjusted from 0.002 to 0.006 m around 3s, making the robot accelerate and eventually overcome the perturbation and walk forward 5s. Desired trajectories are plotted as dashed lines with the same color as the actual variable value.

If the constant perturbation can be detected, R_c can be adjusted to increase robustness. We simulated the push forward with adjustments to R_c , and the robot was able to withstand a constant push of 10.6 N , when $R_c = -0.05\text{ m}$.

4.1.4 Ramp

The robot was set to walk at its nominal gait, but the floor was modified to be a ramp up or down. The ramp inclination was increased or decreased until the robot fell. It was found that the control algorithm can handle positive inclinations very well, but was sensitive to negative ones. Thus, when simulating negative inclinations, we searched for the maximum speed the robot could walk while going down a 13%. This particular value was chosen because the maximum acceptable inclination for accessibility ramps (according to the Brazilian Association of Technical Standards) is 12.5%.

The robot was able to walk upward ramps up to 56% of inclination, while trying to perform the nominal gait, and slowly walk down ramps of -13% incline, with $R_c = -0.0015 \text{ m}$. The simulation results can be seen in Figs. 18 and 19, where the vertical displacement of the CoM can be observed. It can be noted in Fig. 18 that the steps finished prematurely with $r_{foot}^y > 0$, while in Fig 19, the effects of the downhill ramp are mitigated due to the small step lengths.

It is important to note that, for the uphill simulation, the robot had no knowledge about the terrain, and therefore it was not able to change the gait strategy. The controller assumed a flat ground. This resulted in a slower gait when going upwards. Similarly, if R_c was not adjusted, a faster gait would occur when going downwards.

Figure 18: Ramp, 40% incline; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

Figure 19: Ramp, -4% incline; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

4.1.5 Sinusoidal Terrain

For this simulation, the terrain was defined as a function of the position x :

$$y_{Floor}(x) = Amp * \cos(x * Frq) \quad (4.3)$$

Similarly to the downward ramp, the R_c value was decreased to increase robustness. Herein, two simulations with sinusoidal terrain are presented, in the first (Fig. 20), the “frequency” Frq was kept constant at 6, and the “amplitude” Amp was increased until the robot would fall. In the second (Fig. 21), the “amplitude” Amp was kept constant at 0.005 m , and the “frequency” Frq was increased until the robot would fall.

Note that the robot accelerates when descending and decelerates when ascending. The main limiting factor for the robot were the descents, as it can be inferred from 4.1.4, and the maximum viable “amplitude” was $Amp = 0.007 \text{ m}$ for the first case, and the maximum viable “frequency” for the second case was 19.

Figure 20: Sinusoidal terrain $Amp = 0.01\text{ m}$, $Fre = 6$; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

Figure 21: Sinusoidal terrain $Amp = 0.005\text{ m}$, $Fre = 19$; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

4.2 Simulations - Active Fall-Recovery

4.2.1 Simulation Conditions

In this section, two types of simulations will be presented. The first type is similar to the simulations presented previously as they do not include the active fall recovery (FR) module. They will be used to choose adequate parameters and to compare the different actuation requirements during the gait. The second type of simulations are performed with the complete control architecture that includes the active FR.

During simulations, the robot is subjected to a horizontal force applied to its waist. To ensure valid comparison between simulations with different step durations, the force is constant, and imposed during the first 0.1 s of the first step, see Fig. 22. This guarantees that the perturbation occurs always at the same step phase, the beginning of the step.

Figure 22: Force magnitude over time for $Mag = 11.1$

4.2.2 Parameters Evaluation

The relevant reference values, discussed on section 3.2.3 are modified to generate a more robust gait, according to the simulations. The initial reference values for this section are presented in Tab. 7. They yield a gait that is capable of withstanding a push with a magnitude of 6.9 N.

Table 7: Initial reference values

Reference		Value	Unit
Intended step duration	t_{step}	0.4	s
Absolute torso angle	R_γ	0	rad
CoM height	r_{CoM}^{y0}	0.29	m
Swing foot height	R_a	0.01	m
x Distance from swing foot to the CP	R_c	0.001	m

The first modified parameter is r_{CoM}^{y0} , simulation results can be seen in Tab. 8.

Table 8: Modified CoM height r_{CoM}^{y0} - Simulation results

Modified Parameter	Value [m]	Maximum Withstood Magnitude [N]
r_{CoM}^{y0}	0.29	6.9
r_{CoM}^{y0}	0.28	9.2
r_{CoM}^{y0}	0.27	0.7

Note that the hypothesis that a lower CoM position increases robustness is not always true. This hypothesis is based on the simplified LIPM, thus it disregards important characteristics of the complete dynamic of the five links robot. Nevertheless, It is possible to identify a value for r_{CoM}^{y0} that results in a more robust gait. The next simulation results will assume a value for r_{CoM}^{y0} of 0.28 m ; and will vary t_{step} , see Tab. 9.

Table 9: Modified intended step duration t_{step} - Simulation results

Modified Parameter	Value [s]	Maximum Withstood Magnitude [N]
t_{step}	0.35	13.2
t_{step}	0.30	10.1
t_{step}	0.25	6.6

Similarly to the previous result, even though the LIPM model provides a guideline for choosing the parameters, the actual performance can vary largely due to model simplifications. The divergence mainly occurs due to the fact that the LIPM does not include leg masses nor considers friction. Finally the last evaluated parameter is R_c ; Results on Tab. 10.

Changing R_c does not appear to bring any benefits, this occurs because the perturbation caused by the push results in a too long step. Decreasing R_c , on this situation, means taking an even longer step to reach closer to the CP. Thus it is expected that it will not increase the amount of force that can be withstood. But decreasing R_c has an impact on the subsequent steps; and can be used to accelerate the convergence rate to

Table 10: Modified x distance from swing foot to the CP R_c - Simulation results

Modified Parameter	Value [m]	Maximum Withstood Magnitude [N]
R_c	0.0008	13.2
R_c	0.0006	13.2
R_c	0.0004	13.1

the reference gait after the perturbation.

4.2.3 Fall Recovery Implementation

Using the previous simulation results as a guideline, the three fall recovery values can be chosen, see Tab. 11. The standard gait was the initial gait presented in this section (Tab. 7).

Table 11: Fall recovery reference values

FR - Reference		Value	Unit
FR - Intended step duration	t_{step}^{FR}	0.38	s
FR - CoM height	r_{CoM}^{FR-y0}	0.285	m
FR - x Distance from swing foot to the CP	R_c^{FR}	0.0008	m

Note that even though there were better values for the three FR References, a compromise had to be made. Lowering the CoM height decreases the vertical force; and shortening the step duration increases the horizontal force. When both actions are taken at the same time, the robot is subject to slip; thus, a smoother transition is needed. It was also necessary to lower the gain λ_2 from equation 3.10 to 50 instead of 150 (contributing to the smoother transition).

The simulation results can be seen at Fig. 23 and 24. With the active fall recovery enabled, the robot was able to recover from a push of 11, 1 N . The FR strategy generating more robust trajectories was maintained for six steps. This value has been taken so that the disturbance triggers the active FR only once.

4.2.4 Performance vs Energy

To evaluate the effectiveness of switching to the FR strategy if a perturbation is detected, both standard and FR gaits can be analyzed independently. On Fig. 25 and 26 we can see the torques necessary to perform the standard and the FR (more robust) gaits, respectively. In this simulations the active fall recovery is not included and no pushes are

Figure 23: Active fall recovery, trajectories - Simulation results; desired trajectories are plotted as dashed lines with the same color as the actual variable value.

Figure 24: Active fall recovery, torques - Simulation results

applied, it is just a comparison between both gaits. It can be seen that more energy is needed to perform the more robust gait, especially when considering that the more robust gait has a lower speed.

Figure 25: Torques of the standard gait

As presented previously, the standard gait is capable of withstanding 6.9 N , while the more robust one 13.2 N . When the active fall recovery is implemented, the robot can walk most of the time with the energy consumption of the more efficient gait, only transitioning to the more robust one when needed. When it does, it is capable of resisting pushes 60% larger (11.1 N) than the standard gait. Even though the standard gait is not as robust as walking with the FR-parameters all the time, it is a convenient trade off that can be used to increase range for tasks that does not expect large disturbances, eg. walking on a sidewalk or inside buildings.

Figure 26: Torques of the more robust gait

5 DISCUSSION

5.1 Controller

After analyzing the results, we revise the controller requirements in order to see how well they have been met:

- Robustness against perturbations – The controller was based on the capture point/N-step capture region and therefore had a “built in” fall recovery. The simulations showed that the robot could resist different types of perturbations. It was specially successful when going uphill, but it was not as robust for walking on negative slopes.
- Versatile – The controller was developed to easily interact with higher control levels, such as path and task planning. They provide reference values that can directly alter the gait dynamics, resulting in versatile gaits. This was demonstrated in three situations:
 1. speed modulation, presented in section 4.1.2, in this case, the robot was able to dynamically change its velocity;
 2. adaptation to a constant push, as presented in section 4.1.3.1, that shows how the robot is able to overcome the disturbances and keep walking forward;
 3. adaptation to walk on inclined ramps finally by adjustment of R_c as presented in sections 4.1.4 and 4.1.5.
- Low energy consumption – Even though this aspect was not deeply analyzed in this work, the control algorithm allows the optimization of the Bézier curves in order to maximize energy efficiency.
- Real-time implementation – Neither the controller nor the active fall recovery include an optimization or other computationally intensive algorithm, therefore they are adequate for real-time implementation. The most demanding part of the controller is to calculate the Feedback Linearization matrices.

- Generalizable – The Feedback Linearization allows the implementation of the controller in a wide range of biped robots, as long as it is possible to control a set of relevant gait variables. Extending the algorithms to 3D is necessary for the implementation in robots that have a 3D working space.

Nevertheless, there were some drawbacks that negatively impacted performance: Note that in the nominal or standard gait (see Fig. 11 in section 4.1.1), the prediction of the CP at the end of the step $r_{CP}(t_{step})$ changes significantly during the step. This happens because the LIPM model does not consider mass nor moment of inertia on the leg; bringing the leg forward during the step accelerates the CP, thus the foot tends to “chase” the CP in order to reach the desired distance from the CP (D_c). This causes a high foot x velocity at impact, which increases the risk of slip, and reduces the accuracy of the actual distance between the CP and the swing foot before the impact, which influences directly the robot stability.

Moreover, the calculated capture regions were much larger than ones achieved during simulations. The maximum calculated capture region radius to the nominal gait was 0.4654 m ; while the actual viable largest capture region recorded on section 4.1.2 was 0.004 m . We believe this occurred not only due to the differences between the LIPM and the five links model, but also due to an inaccurate value of “Largest Possible Step” L_{max} . L_{max} was considered to be the maximum reachable step possible based on leg dimensions. However, during simulations, taking a too large step (still smaller than the maximum reachable step) resulted in slip, therefore, we believe L_{max} should be corrected to the maximum dynamic possible step, and the calculated capture regions (for this specific robot/application) reevaluated. Note that the maximum dynamic step will vary depending on the gait.

The gait control, both nominal and FR, can be improved with the implementation of a systematic procedure to determine and optimize gains, Bézier curves and reference values. These are responsible to guarantee not only good energy efficiency, but also gait stability. Since the reference values are directly related to the gait patterns, adjustments can be made in a straightforward way. Nevertheless, for a proper implementation on a real robot, a thorough analysis of these values must be made.

Finally, it can be noted that the controller allowed the pointed feet robot to walk successfully under different perturbations, on the 2-Step capture region.

5.1.1 Active Fall Recovery

The active fall recovery was implemented to change the desired trajectories and increase gait robustness when needed. It was possible to switch to a more robust gait when a perturbation was detected, and when not needed, the robot maintained more energy efficient normal gait.

The FR implementation allowed a 60% increase in the force the robot was able to withstand (in the presented scenario). To achieve this results, the desired trajectories were modified in three ways: lowering the CoM and subsequently the hip height, decreasing the step duration and finishing the step closer to the CP. Similar to the control algorithm, the FR success relies on the adequate choice of reference values, gains and parameters, and could be improved with systematic optimization of those.

Depending on the standard gait, the abrupt change of the desired trajectories to more robust ones may cause the robot to fall, so the FR reference values must be adjusted for each individual standard gait.

5.2 Five Links Model & Robot

The proposed control algorithm was tested with several simulation of a five links, 2D biped robot model. In order to have a better understanding of how the controller will perform when implemented on the physical robot, we must consider the possible discrepancies between the physical robot and the model:

- Joint friction – The values used on the simulations for viscous friction on the robots joints were based on the values reported for RABBIT (CHEVALLEREAU et al., 2003). However, the actuators used in the robot Bipo are quite different. Also, any dry friction present on the bearings or gearboxes was not considered. A better assessment of the actual values must be made to achieve a successful implementation of the feedback linearization. The actuators of Bipo, manufactured by Maxon, are very efficient and have low gear ratio, thus mitigating this issue.
- 3D robot and Stabilization Bar – The designed controller assumes a 2D robot, which is not possible for a real physical robot. The real robot will use a stabilization bar to restrict its motion to the sagittal plane and walk in circles. This “simplification” overlooks some aspects of the actual dynamics, eg.: the inertia of the stabilization bar, the dynamic coupling of the 3D mechanism and the fact that, instead of walking

on a plane, the robot will be walking on a “sphere”. Some of these issues can be mitigated with the design, such as the addition of counter weights and wheels at the feet. Nevertheless, this will have a impact on gait performance. Considering similar solutions have already been proven to be successful (CHEVALLEREAU et al., 2003), we believe this issues can be overcome.

- Inertia and mass distribution – The Feedback Linearization (FL) relies on an accurate model of the robot to be properly implemented, Inertia and other robot properties were obtained from the detailed 3D model of the robot. However, some parameters are difficult to determine accurately, such as the mass distribution of the servomotors, mass and inertia of cables and cable slacks (that are necessary to guarantee the range of motion). To overcome these problems, the control algorithm will depend on the PD gains that are implemented “on top” of the FL.
- Sensor and actuation noise and delay – The simulations did not consider the real behaviour of sensors and actuators. Nevertheless, tests with the real robot will determine how much these flaws affect the gait.
- Impact dynamics – In the simulations, the impact was assumed to be instantaneous and non-elastic, based on the formulation proposed by Westervelt et al. (WESTERVELT et al., 2007). However, it does not represent perfectly the actual dynamics, and thus will have an effect on the gait pattern. Since the gait trajectories are adapted at the beginning of the step, this divergence may not be an issue.
- Impact detection – A proper sensor must be implemented to identify that the swing foot has reached the ground. Switches, accelerometers, pressure sensors could be alternatives to do so. If the contact is not detected properly, the robot will try to move the foot as if it were still off the ground, resulting in unexpectedly pushing the ground and destabilizing the robot.
- Foot and ground interaction – On the simulations, the foot/ground interaction was modeled as a perfect rotational joint. On the real robot, the robot will rotate around its foot, which will not behave as a perfect joint and it has some friction. The friction may help slow the under-actuated dynamic, but the imperfect joint might have a negative impact on gait performance.
- “Transparent” gearbox/backdrivability – The robot design does not include force sensors on the joints, therefore the controller relies on the current control of the servomotor to apply the desired torques to the joints. Therefore, a low gear ratio

was chosen, yielding a more “transparent” gearbox for accurate torque control, and allowing backdrivability for proper impedance control.

Despite the limitations presented above, we believe that the gait controller presented here, with minor adjustments, is adequate for implementation on the real robot:’

- Actuation requirements – The simulations showed that the maximum torques were well within the stall torque limit of the selected motors during regular walking. Moreover, the required torque peaks during perturbed gait were also feasible.
- Real time constraints – The presented algorithm is a straightforward feedback implementation, and does not require neither optimization nor other computationally demanding routines. Therefore, it is adequate for real time implementation.
- Simulation results – It can be observed in the simulations that the algorithms are capable of sustaining gait with some robustness. This robustness will compensate for some of the limitations presented above.

6 CONCLUSION

In this work we developed a biped gait controller with active fall recovery along with the complete design of a biped robot. The biped robot represents walking on the sagittal plane, with a design that lowers costs and facilitates manufacturing. The gait controller is based on the capture point/N-step capturability stability criterion theory, which is commonly used for fall recovery strategies. Therefore, it not only has built-in passive fall recovery capabilities, but also allows a straightforward implementation of an active Fall Recovery algorithm. The controller purpose is to produce reliable walking with robustness against perturbations, simulation showed promising results; while the FR algorithm was able to significantly increase robustness against perturbations.

6.1 Future Work

The results obtained are promising enough to guarantee pursue further research:

- Develop a systematic way to optimize the gait patterns – The proposed algorithms are sensitive to the reference values, parameters and gains, thus obtaining a procedure to obtain optimal (or even sub-optimal) values for these parameters is essential to maximize energy efficiency and robustness. We believe that optimization may be a resourceful tool to improve the controller and FR performance, especially in the less successful tasks.
- Build the designed robot – This will allow us to conduct validation tests of the proposed algorithms with a physical device. This will also provide a better understanding of their qualities, flaws and therefore unveiling ways for improvement. This will also allow for further research on other control algorithms and aspects of legged locomotion.
- Extend the proposed algorithms to 3D – Despite of the simplification made of representing the gait dynamics in the sagittal plane, useful robots are not 2D. In this

way, the choice of the capture point/ N step capturability theory was made with the knowledge that the extension to the 3D is straightforward. This will be essential to develop versatile robots capable of performing several tasks.

Thank you,
—Ivan

REFERENCES

- CHEVALLEREAU, C. et al. Rabbit: A testbed for advanced control theory. *IEEE Control Systems Magazine*, p. 57–79, 2003.
- CONCONI, A. et al. *Mapas e gráficos mostram avanço do novo coronavírus no mundo e no Brasil*. 2020. Disponível em: <<https://www.estadao.com.br/infograficos/saude, mapas-e-graficos-mostram-avanco-do-novo-coronavirus-no-mundo-e-no-brasil,1085946>>.
- CORDERO, A. F.; KOOPMAN, H. J. F. M.; Van Der Helm, F. C. T. Mechanical model of the recovery from stumbling. *Biol. Cybern*, v. 91, p. 212–220, 2004.
- DA, X.; HARTLEY, R.; GRIZZLE, J. W. Supervised learning for stabilizing underactuated bipedal robot locomotion, with outdoor experiments on the wave field. *Proceedings - IEEE International Conference on Robotics and Automation*, p. 3476–3483, 2017. ISSN 10504729.
- DOLZAN, M. *Monitor acompanha taxas de isolamento social no Brasil*. 2020. Disponível em: <<https://www.estadao.com.br/infograficos/saude, monitor-acompanha-taxas-de-isolamento-social-no-brasil,1093828>>.
- DUYSENS, J.; FORNER-CORDERO, A. Walking with perturbations: a guide for biped humans and robots. *Bioinspiration & Biomimetics*, IOP Publishing, v. 13, n. 6, p. 061001, sep 2018. ISSN 1748-3190.
- DUYSENS, J.; FORNER-CORDERO, A. *A controller perspective on biological gait control: Reflexes and central pattern generators*. [S.l.]: Elsevier Ltd, 2019.
- Ford Motor Company; Agility Robotics. *Digit: Future of Self-Driving Vehicle Delivery / The Future of Ford and Transportation / Ford - YouTube*. 2019. Disponível em: <<https://www.youtube.com/watch?v=WHWciIXNK2c>>.
- GARCIA, G.; GRIFFIN, R.; PRATT, J. 0-Step Capturability, Motion Decomposition and Global Feedback Control of the 3D Variable Height-Inverted Pendulum. dec 2019.
- GOSWAMI, A.; ESPIAU, B.; KERAMANE, A. Limit Cycles in a Passive Compass Gait Biped and Passivity-Mimicking Control Laws. *Autonomous Robots*, Kluwer Academic Publishers, v. 4, n. 3, p. 273–286, 1997. ISSN 09295593.
- GRIFFIN, R. J.; LEONESSA, A. *Model Predictive Control for Stable Walking Using the Divergent Component of Motion with Footstep Location and Yaw Adaptation*. [S.l.: s.n.], 2019. v. 16. 1–37 p. ISSN 02198436. ISBN 0219843619.
- GRIFFIN, R. J. et al. Walking stabilization using step timing and location adjustment on the humanoid robot, Atlas. *IEEE International Conference on Intelligent Robots and Systems*, v. 2017-Septe, p. 667–673, 2017. ISSN 21530866.

- GRIZZLE, J.; ABBA, G.; PLESTAN, F. Asymptotically stable walking for biped robots: analysis via systems with impulse effects. *IEEE Transactions on Automatic Control*, v. 46, n. 1, p. 51–64, 2001. ISSN 00189286.
- GRIZZLE, J. W. et al. MABEL, A New Robotic Bipedal Walker and Runner. *IEEE Transaction on Automatic Control*, v. 54, n. 8, Aug. 2009, p. 1751 – 1764, 2009.
- HOF, A.; GAZENDAM, M.; SINKE, W. The condition for dynamic stability. *Journal of Biomechanics*, Elsevier, v. 38, n. 1, p. 1–8, jan 2005. ISSN 0021-9290.
- HONG, Y.-D.; KIM, J.-H. 3-D Command State-Based Modifiable Bipedal Walking on Uneven Terrain. *IEEE/ASME Transactions on Mechatronics*, v. 18, n. 2, p. 657–663, apr 2013. ISSN 1083-4435.
- IJSPEERT, A. J. Central pattern generators for locomotion control in animals and robots: A review. *Neural Networks*, Pergamon, v. 21, n. 4, p. 642–653, may 2008. ISSN 0893-6080.
- KAJITA, S. et al. The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. In: *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*. [S.l.]: IEEE, 2001. v. 1, p. 239–246. ISBN 0-7803-6612-3.
- KAJITA, S. et al. Biped walking pattern generation by using preview control of zero-moment point. In: *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*. [S.l.]: IEEE, 2003. p. 1620–1626. ISBN 0-7803-7736-2.
- KOOLEN, T. et al. Capturability-based analysis and control of legged locomotion, Part 1: Theory and application to three simple gait models. *The International Journal of Robotics Research*, v. 31, n. 9, p. 1094–1113, 2012.
- KOOLEN, T.; POSA, M.; TEDRAKE, R. Balance control using center of mass height variation: Limitations imposed by unilateral contact. In: *IEEE-RAS International Conference on Humanoid Robots*. [S.l.]: IEEE Computer Society, 2016. p. 8–15. ISBN 9781509047185. ISSN 21640580.
- MANOONPONG, P. et al. Adaptive, fast walking in a biped robot under neuronal control and learning. *PLoS Computational Biology*, Public Library of Science, v. 3, n. 7, p. 1305–1320, 2007. ISSN 15537358.
- MCGEER, T. Passive Dynamic Walking. *The International Journal of Robotics Research*, Sage PublicationsSage CA: Thousand Oaks, CA, v. 9, n. 2, p. 62–82, apr 1990. ISSN 0278-3649.
- PRATT, J. et al. Capture point: A step toward humanoid push recovery. In: *Proceedings of the 2006 6th IEEE-RAS International Conference on Humanoid Robots, HUMANOIDS*. [S.l.: s.n.], 2006. p. 200–207. ISBN 142440200X. ISSN 2164-0572.
- PRATT, J. et al. Capturability-based analysis and control of legged locomotion, Part 2: Application to M2V2, a lower-body humanoid. *The International Journal of Robotics Research*, v. 31, n. 10, p. 1117–1133, 2012.

RAMOS, J. et al. Facilitating Model-Based Control Through Software-Hardware Co-Design. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. [S.l.]: IEEE, 2018. p. 566–572. ISBN 978-1-5386-3081-5.

ROSSI, L. F. *Desenvolvimento e implementação de um algoritmo bioinspirado para o controle de marcha em robôs bípedes*. Tese (Doutorado) — Universidade de São Paulo, 2017.

ROSSI, L. F. et al. Predicted Step Viability: a stability criterion for biped gait. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Springer Verlag, v. 41, n. 12, dec 2019. ISSN 18063691.

SLOTINE, J.-J. E. J.-J. E.; LI, W. *Applied nonlinear control*. Englewood Cliffs N.J.: Prentice Hall, 1991. 459 p. ISBN 9780130408907.

TAKENAKA, T.; MATSUMOTO, T.; YOSHIIKE, T. Real time motion generation and control for biped robot - 1st report: Walking gait pattern generation-. In: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. [S.l.]: IEEE, 2009. p. 1084–1091. ISBN 978-1-4244-3803-7.

VUKOBRATOVIĆ, M.; BOROVIAC, B. Zero-moment point — thirty five years of its life. *International Journal of Humanoid Robotics*, v. 01, n. 01, p. 157–173, mar 2004. ISSN 0219-8436.

WESTERVELT, E.; GRIZZLE, J.; KODITSCHKEK, D. Hybrid zero dynamics of planar biped walkers. *IEEE Transactions on Automatic Control*, v. 48, n. 1, p. 42–56, jan 2003. ISSN 0018-9286.

WESTERVELT, E. R. et al. *Feedback Control of Dynamic Bipedal Robot Locomotion*. [S.l.]: CRC press, 2007.

WHO. *Coronavirus disease (COVID-19) advice for the public*. 2020. Disponível em: <<https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public>>.

WRIGHT, J.; JORDANOV, I. Intelligent Approaches in Locomotion - A Review. *Journal of Intelligent & Robotic Systems*, v. 80, n. 2, p. 255–277, nov 2015. ISSN 0921-0296.

APPENDIX A – PROJECT BIPO

Designing a biped robot is one of the main goals of this dissertation. The robot is named Bipo, when built, It will serve as a tool for legged locomotion research in USP. It will allow the validation of the control strategies developed inside the Biomechchatronics Laboratory, as the one presented herein and the one developed by Rossi and Forner-Cordero (ROSSI et al., 2019). It will also be used to compare different controllers of the current state of the art on bipedal locomotion.

Bipo’s design was strongly influenced by RABBIT (CHEVALLEREAU et al., 2003). We designed it to achieve similar performance with low cost and allowing it to be manufactured with resources available in USP. In this chapter we present the key aspects of the mechanical and electronic design of Bipo. During the design, the Software-Hardware Co-Design philosophy introduced by Ramos et al. (RAMOS et al., 2018) was used.

A.1 Mechanical Design

As presented on Chapter 3, bipedal gait has been simplified to the sagittal plane. To represent a 2D model with a 3D robot, the lateral stability must be ensured by another structure called stabilization bar or “boom” similarly to RABBIT and MABEL (CHEVALLEREAU et al., 2003; GRIZZLE et al., 2009). Frontal stability is ensured by the control algorithm.

The five links model (presented on chapter 3.1.2) is the base for the mechanical design of Bipo. This model may be extended to a seven links one, where feet are added. The 2D Model allows a simplification of the Bipo’s structure; a modular approach was chosen to obtain a simple design, where each degree of freedom is a module. The modules were made to be as simple as possible to reduce the number of fail points and facilitate manufacturing.

A module can be seen in Fig. 27a. It consists of a motor+reduction, a pulley to connect to the previous module, an aluminum structure and an axis+bearings to connect

to the next module. The motor actuates the next module through belt drive, the joints are not directly driven to more conveniently distribute the weight. On the stance leg, its optimal to have the majority of the weight as far from the feet as possible, resulting in a higher inertia (relative to the ankle) and slower fall; on the swing leg, its best to have a lower inertia (relative to the hip) so the leg can swing forward faster, so most of the weigh should be near the hip. To achieve this weight distribution, the motor is pushed upwards and actuation occurs through belt transmission. Besides the structure modules, there is the electronics module that can be seen in Fig. 27b.

Figure 27: Modules

The five links robot is composed of 4 active structural modules, 2 passive, and 2 electronics modules. It can be seen in Figs. 28, 29a and 29b.

On the passive modules we can observe the point feet that makes contact with the ground. Like on RABBIT and MABEL (CHEVALLEREAU et al., 2003; GRIZZLE et al., 2009) the point feet are wheels with the rotating axis parallel to the sagittal plane. This is due to two reasons:

- Extending the leg forward or backwards increases the radius of the circular trajectory imposed by the stabilization bar, this translates into a lateral displacement of the contact point on the ground. see Fig. 30.
- If the robot extend or contracts its legs (in the vertical direction), it pivots around a fixed point of the stabilization bar, this results in a rotation on the frontal plane and subsequently a lateral displacement of the contact point on the ground, see Fig 31.

Figure 28: Bipo - Five links model

Even though these displacements are relatively small, to allow cinematic compatibility the wheels are necessary.

To extend the robot from five links to seven, it is necessary to change both passive modules to active ones, and add non pointy feet. Herein a preliminary version of the seven links is presented, see Fig. 32. The cinematic incompatibilities presented before haven't been solved for the seven links yet.

All the critical components have been dimensioned to ensure mechanical robustness. On Fig 33 it can be seen the finite elements analysis of the pulley. Due to the complex shape of this part, it was designed (and dimensioned) to be manufactured by 3D printing. Other structural elements have also been dimensioned, the complete list can be found in table 12

Table 12: Structural sizing, parts list - Bipo

Part	Evaluated Condition
Bearings	Static load
Module's aluminum plate	Bending moment
Module's axis	Shear stress
Contact between plastic pulley and aluminum	Friction force

The material chosen for most of the structure was the aluminum alloy 5052, its rea-

Figure 29: Bipo - Five links model, front and side view

Figure 30: Wheel as the point feet - Cinematic incompatibility 1

(a) Robot is presented in two positions, solid and transparent

(b) Top view, contact points as red crosses, circular trajectory and lateral displacement (d)

Figure 31: Wheel as the point feet - Cinematic incompatibility 2

- (a) Robot is presented in two positions, crouching (transparent) and standing up (solid)
 (b) Detail of feet positioning and lateral displacement d , red crosses indicate the contact points

Figure 32: Bipo - Preliminary seven links model

Figure 33: Finite elements analysis - Pulley

sonably priced and has a good strength to weight ratio. Stainless steel SAE 420 was chosen for the axis due to its good mechanical properties, also, being stainless results in easier maintenance during its working life at the Laboratory of Biomechatronics.

A.1.1 Stabilization Bar

To restrict Bipo's movement to the sagittal plane, and ensure lateral stability, it is necessary to add another mechanical structure, the stabilization bar.

Figure 34: Boom assembly

The 3D space has 6 degrees of freedom 3 rotational and 3 linear, while the 2D has

only 1 rotational and 2 linear. Therefore the stabilization bar must have only 3 DoF; they are all rotational, but their combination allows movement as if the robot was restricted to a spherical shell. If the radius is big enough, the robot will walk as if it was restricted to the sagittal plane. The stabilization bar and its DoF can be seen in Fig. 34.

To ensure mechanical robustness, its components have been mechanically sized as listed on table 13.

Table 13: Structural sizing, parts list - Stabilization bar

Part	Evaluated Condition
Counterbalance	Mass
Bearings	Static load
Axes	Bending moment
Beam	Bending moment
Aluminum structural parts	Bending moment

A.1.2 Actuators

The first design choice focused on the actuators, they have to be sized properly and they can be used to define the model scale. There is also a limited set of options due to cost and commercial availability. In standard applications the actuators doesn't need to support their own weight, for example, a car's engine and structure is supported by the suspension. But in a legged robot the actuators must withstand the whole weight; therefore the power to weight and torque to weight ratios are extremely important.

One option to choose is hydraulic actuators, since their force/torque to weight can be high; but accurate force control for this type of actuator is extremely difficult, maintenance is not as convenient as other options and energy consumption presents a challenge for autonomy; in spite of that, Atlas from Boston Dynamics uses hydraulic actuation. Keeping in mind the software-hardware co-design philosophy (RAMOS et al., 2018), the electric motors alternative is more suitable for the proposed application.

The only goal for this robot is to serve as a test platform, therefore the trade off between total payload and controllability is well worth. Since the robot is stationary, it is not necessary for it to carry its batteries or other heavy components; this will allow us to lower the costs, without interfering with the desired results.

After these considerations we can properly set the requirements for the actuators:

- Controllability – Actuators must allow accurate position and force control to ensure

any control strategy can be implemented;

- High torque/power to weight ratio – As mentioned above, this characteristic is essential to legged robots (this ratio will be further explored in the next subsection);
- High efficiency – To allow longer run time;
- Easy/low cost maintenance – This will ensure the robot has a low down time for repairs.

Based on this requirements electric DC motors where chosen, combined with the correct drivers, they allows the necessary controllability and efficiency, they are also easy to implement and maintain. To ensure compatibility between components it was decided to acquire de motor, drive, encoder and gear box from the same manufacturer, Maxon.

A.1.2.1 Motor Sizing

To size the motor, an iterative method has to be used. This occurs because its necessary to find the correct balance between robot weight (witch is very influenced by the motor's weight) and motor torque. Two critical scenarios where used:

- Standing up from a “squat” is the critical condition for the knee joint, see Fig. 35a;
- Swinging the swing leg forward the hip joint is the critical joint, see Fig. 35b.

Based on the first critical scenario, a table can be made to define the necessary torque to weight ratio. Its assumed as a simplifying hypothesis that the mass of the robot is proportional to the mass of the motor (m_m) by a factor p , therefore for a given motor the total weight that needs to be supported is $p * m_m * g$ (g is the gravitational constant).

If we consider $\theta = 0$, a gear ration r and a “femur” length l to total torque τ provided by the motor must be:

$$\tau = \frac{m_m * p * g * l}{r} \quad (\text{A.1})$$

We can write the ratio torque to mass ratio as a function of the length l and the gear ratio r :

$$\frac{\tau}{m_m} = \frac{p * g * l}{r} \quad (\text{A.2})$$

Figure 35: Critical scenarios for motor sizing

(a) Standing up from a “squat”

(b) Swinging the sing leg forward

$\frac{\tau}{m_m}$ is a convenient way to compare motors of different sizes. Varying l and r changes the necessary $\frac{\tau}{m_m}$ value. Table 14 presents a range of “femur” sizes and gear ratios. For a torque to weight threshold one can choose a combination of gear ratio and leg length. Note that, for simplicity, the robots mass does not change when the leg size changes, witch is not true for the real robot.

This table allowed us to determine the basic dimensions of the robot. The total leg length should be around 150 mm , the gear ratio should be close to 15:1 and the motor torque to mass ratio should be above 6.5 mNm/g . A low gear ratio was chosen to have a better dynamic response during the impact, allow backdrivability, and also improve the force control on the joint.

The second critical scenario was used to analyze the time it took the motor to swing the swing leg forward. As it can be concluded from eq. 3.4:

$$r_{CP}(t) = [r_{CP}(0) - r_{ankle}]e^t + r_{ankle} \quad (A.3)$$

If the robot is unbalanced (large $r_{CP}(0)$), It will be able to recover its balance by stepping in the direction of the capture point. If the robot is able to take large and fast steps it will be able to recover from larger unbalances than otherwise. Both cases presented so

Table 14: Torque to weight vs gear ratio and leg length

Torque Density											
-	-	Femur Lenght									
-	-	0,05	0,075	0,1	0,125	0,15	0,175	0,2	0,4	0,45	1
Gear Ratio	1	32,46	48,69	64,92	81,15	97,39	113,62	129,85	259,69	292,16	649,23
	2	16,23	24,35	32,46	40,58	48,69	56,81	64,92	129,85	146,08	324,62
	3	10,82	16,23	21,64	27,05	32,46	37,87	43,28	86,56	97,39	216,41
	4	8,12	12,17	16,23	20,29	24,35	28,40	32,46	64,92	73,04	162,31
	5	6,49	9,74	12,98	16,23	19,48	22,72	25,97	51,94	58,43	129,85
	6	5,41	8,12	10,82	13,53	16,23	18,94	21,64	43,28	48,69	108,21
	7	4,64	6,96	9,27	11,59	13,91	16,23	18,55	37,10	41,74	92,75
	8	4,06	6,09	8,12	10,14	12,17	14,20	16,23	32,46	36,52	81,15
	9	3,61	5,41	7,21	9,02	10,82	12,62	14,43	28,85	32,46	72,14
	10	3,25	4,87	6,49	8,12	9,74	11,36	12,98	25,97	29,22	64,92
	11	2,95	4,43	5,90	7,38	8,85	10,33	11,80	23,61	26,56	59,02
	12	2,71	4,06	5,41	6,76	8,12	9,47	10,82	21,64	24,35	54,10
	13	2,50	3,75	4,99	6,24	7,49	8,74	9,99	19,98	22,47	49,94
	14	2,32	3,48	4,64	5,80	6,96	8,12	9,27	18,55	20,87	46,37
	15	2,16	3,25	4,33	5,41	6,49	7,57	8,66	17,31	19,48	43,28
	16	2,03	3,04	4,06	5,07	6,09	7,10	8,12	16,23	18,26	40,58
	17	1,91	2,86	3,82	4,77	5,73	6,68	7,64	15,28	17,19	38,19
	18	1,80	2,71	3,61	4,51	5,41	6,31	7,21	14,43	16,23	36,07
	20	1,62	2,43	3,25	4,06	4,87	5,68	6,49	12,98	14,61	32,46
	30	1,08	1,62	2,16	2,71	3,25	3,79	4,33	8,66	9,74	21,64
	40	0,81	1,22	1,62	2,03	2,43	2,84	3,25	6,49	7,30	16,23
	50	0,65	0,97	1,30	1,62	1,95	2,27	2,60	5,19	5,84	12,98
	60	0,54	0,81	1,08	1,35	1,62	1,89	2,16	4,33	4,87	10,82
	70	0,46	0,70	0,93	1,16	1,39	1,62	1,85	3,71	4,17	9,27
	80	0,41	0,61	0,81	1,01	1,22	1,42	1,62	3,25	3,65	8,12
	90	0,36	0,54	0,72	0,90	1,08	1,26	1,44	2,89	3,25	7,21
	100	0,32	0,49	0,65	0,81	0,97	1,14	1,30	2,60	2,92	6,49

far have considerable simplification; hence, the selection had to be validated by a more extensive analysis. Despite that, they provided enough information to make a preliminary choice of components:

- Motor – Maxon EC 45 flat, brushless, 70 watt;
- Drive – Maxon EPOS4 compact 50/5 CAN, digital positioning controller, 5 A, 10 - 50 VDC;
- Encoder – Maxon encoder MILE, 2048 CPT, 2 channels;
- Gear Box – Maxon planetary gearhead GP 42, 3 - 15 Nm, ceramic version.

The motor was chosen among Maxon’s EC Flat family of motors, due to it’s high torque to weight ratio. With this components, the complete design was completed as presented on section A.1.

Based on the robots properties, the simulations presented in the section 4 were conducted, confirming the motor selection was adequate. The simulations results were analyzed to determine the demands for the actuators, e.g. Fig. 36, that presents the joints torques (τ^j) for normal walking on a plane.

Figure 36: Torque vs time for walking on a plane

Moreover, the design allows adjustments to the belt reduction if needed (requires some modifications).

A.2 Electronics Design

The electronics design must be compatible to the control strategies and mechanical design of Bipo. Therefore the requisites that guide the electronic structure and components selections can be set:

- Drivers must allow force control;
- The onboard processor must have computational capability to run the control algorithms at high rate (frequency $> 250 \text{ Hz}$);
- The electronic structure must allow modularity to add actuators and sensors.

Figure 37: Electronic diagram

A modular structure was chosen, see Fig. 37, it can be subdivided into two levels. The lower level utilizes CANopen as communication protocol, that allows the addition of sensors and actuators. The higher level utilizes Ethernet as communication protocol. The communication between devices is managed by the Robot Operating System (ROS), ROS has a vast collection of tools and libraries for robots and is widely used for similar applications.

The chosen onboard processor is Raspberry Pi, due to its low cost and high computational capability. If necessary the structure allows the addition of extra Raspberries Pi

to increase the computational power. The inclusion of a desktop/laptop was also foreseen to manage data acquisition.

The Raspberry Pi does not have CAN capabilities natively, therefore the inclusion of a PiCAN2 CAN-Bus Board is necessary. A encoder is needed to determine the torso orientation, the absolute encoder from CUI Inc AMT212D-V was chosen. It uses RS 485 as communication protocol, therefore it was necessary to include a converter module.

As presented in the previous section the drive EPOS4 Compact 50/5 CAN was chosen due to its capability of current control and compatibility with the motor.