



Universidad  
Carlos III de Madrid

DEPARTAMENTO DE INGENIERÍA DE SISTEMAS Y AUTOMÁTICA

## TESIS DE MÁSTER

### BALANCE CONTROL OF HUMANOID ROBOT TEO USING FORCE/TORQUE SENSORS

Autor: María Dolores Pinel del Valle

Tutor: Juan Miguel García Haro

Director: Santiago Martínez de la Casa Díaz

MÁSTER OFICIAL EN  
ROBÓTICA Y AUTOMATIZACIÓN

LEGANÉS, MADRID

JULIO 2016



UNIVERSIDAD CARLOS III DE MADRID  
MÁSTER OFICIAL EN ROBÓTICA Y AUTOMATIZACIÓN

El tribunal aprueba la Tesis de Máster titulada "**Balance control of humanoid robot TEO using Force/Torque sensors**" realizada por **María Dolores Pinel del Valle**.

Fecha: Julio 2016

Tribunal: \_\_\_\_\_  
Dr. Ramón Barber Castaño

\_\_\_\_\_  
Dr. Alberto Jardón Huete

\_\_\_\_\_  
Dra. Concepción Alicia Monje Micharet



*A Jose,  
por cambiar el argumento de mi vida.*

*A mis padres,  
por animarme a llegar hasta aquí.*

*Y por siempre,  
David.*



# Contents

<b>Index of Tables</b>	<b>xi</b>
<b>Index of Figures</b>	<b>xiii</b>
<b>List of Abbreviations and Symbols</b>	<b>xv</b>
<b>Acknowledgments</b>	<b>xix</b>
<b>Resumen</b>	<b>xxi</b>
<b>Abstract</b>	<b>xxiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation and origin of the Thesis . . . . .	1
1.2 Objectives . . . . .	2
<b>2 Literature review</b>	<b>5</b>
2.1 Trends in humanoid robotics . . . . .	5
2.1.1 General classification of robots . . . . .	6
2.1.2 Humanoid robots . . . . .	11
2.2 Bipedal locomotion . . . . .	19
2.3 Biped balance/equilibrium . . . . .	22

2.4	Zero Moment Point (ZMP) . . . . .	23
2.4.1	Equations of ZMP . . . . .	25
2.4.2	Relation between COG and ZMP . . . . .	28
2.4.3	ZMP areas. . . . .	31
<b>3</b>	<b>Platform description</b>	<b>33</b>
3.1	Humanoid robot TEO . . . . .	33
3.2	Force/Torque sensors . . . . .	35
3.3	Data Acquisition . . . . .	37
3.3.1	Acquisition program . . . . .	38
<b>4</b>	<b>Control Architecture</b>	<b>41</b>
4.1	Introduction . . . . .	41
4.2	Inverted Pendulum Model . . . . .	42
4.3	Feedback in state space. The Linear Quadratic Regulator . . . . .	47
4.4	Stabilizer . . . . .	52
4.5	Control strategies . . . . .	53
<b>5</b>	<b>Experimental results</b>	<b>57</b>
5.1	Ankles angle - ZMP relation . . . . .	58
5.2	Position control . . . . .	59
5.3	Velocity control . . . . .	61
5.4	ZMP error compensation . . . . .	62
5.4.1	Error compensation model . . . . .	63
5.4.2	Experimental error acquisition . . . . .	65
5.4.3	ZMP error matching . . . . .	66
5.5	Compensated position and velocity control . . . . .	67
5.6	ZMP areas . . . . .	70
5.7	Push-Recovery experiments . . . . .	73

<b>6 Conclusions and future work</b>	<b>75</b>
6.1 Conclusions . . . . .	75
6.2 Future work . . . . .	77
<b>Bibliography</b>	<b>79</b>
<b>Appendix A JR3 manufacturer information</b>	<b>83</b>
<b>Appendix B JR3 85M35A sensor drawings</b>	<b>87</b>
<b>Appendix C Foot drawings</b>	<b>91</b>
<b>Appendix D Ankle motors</b>	<b>95</b>
<b>Appendix E Guide to the code</b>	<b>99</b>



# Index of Tables

3.1 F-T sensor models and characteristics. [JR3 Inc.] . . . . .	36
5.1 ZMP comparison using Classical and Proposed LIMP values . . . . .	69



# Index of Figures

2.1	ABB industrial manipulator . . . . .	7
2.2	Mobile robots . . . . .	7
2.3	Zoomorphic robots . . . . .	8
2.4	Humanoid robots . . . . .	9
2.5	Uncanny valley . . . . .	10
2.6	Hybrid robots . . . . .	10
2.7	Leonardo da Vinci's mechanical knight . . . . .	11
2.8	University of Waseda WL robots . . . . .	12
2.9	WABOT series . . . . .	13
2.10	WABIAN series . . . . .	13
2.11	Honda humanoid prototypes. From left to right: E0 (1986) , E1(1987), E3(1989), E5(1992), P1(1993), P2(1996), P3(1997), ASIMO(2000) . .	14
2.12	HRP series . . . . .	15
2.13	(a) HUBO robot (b)Albert HUBO robot . . . . .	16
2.14	Boston Dynamics humanoid robots . . . . .	17
2.15	Mini humanoids robots . . . . .	18
2.16	REEM-C robot from PAL Robotics . . . . .	18
2.17	Rh-1 humanoid robot . . . . .	19
2.18	Axes division of human body. . . . .	20

2.19	Half cycle phases of biped walking.	21
2.20	Phases of foot support during a walk.	22
2.21	Support areas depending on the support type.	23
2.22	Forces acting on the foot of the bipedal mechanism (Vukobratović & Borovac, 2004)	25
2.23	3D Linear Inverted Pendulum with a contact polygon (Kaynov, 2008)	28
2.24	Single inverted pendulum model	31
2.25	ZMP stability regions in single-support.	32
3.1	(a) Robot TEO and (b) joint distribution diagram	34
3.2	Basic communications diagram	35
3.3	JR3 model 85M35A Force-Torque sensor	36
3.4	JR3 data acquisition card	37
3.5	Data acquisition diagram	38
3.6	Screenshot of the data acquisition interface	39
3.7	Python interface for Single and Double Support with delimited ZMP areas	40
4.1	Single inverted pendulum.	43
4.2	Single inverted pendulum with compliant joint.	44
4.3	LQR controller block diagram	49
4.4	Simulation of inverted pendulum with initial conditions. $\theta(0) = 0.08rad$	49
4.5	ZMP LQR control system.	50
4.6	Simulation of linear inverted pendulum step response.	51
4.7	Simulation of linear inverted pendulum smooth step response.	51
4.8	Stabilizer architecture.	52
4.9	Recovery strategies.	54
4.10	Influence of hip and ankle angles in the frontal plane stability.	55

5.1	Experimental setup of robot TEO . . . . .	57
5.2	Experimental $\theta_A - ZMP_{FT}$ relation . . . . .	58
5.3	Position control diagram . . . . .	59
5.4	Step responses with measured ZMP error in joint position control	60
5.5	Velocity control diagram . . . . .	62
5.6	Step responses with measured ZMP error in joint velocity control	62
5.7	Error compensation diagram . . . . .	63
5.8	Proposed Spring-Damper-Mass Inverted Pendulum Model . . . . .	64
5.9	Reference ZMP and measured ZMP comparative . . . . .	66
5.10	Position control step responses . . . . .	68
5.11	Velocity control step responses . . . . .	68
5.12	ZMP comparison using Classical and Proposed LIMP . . . . .	69
5.13	Mechanical distribution of the robot's ankle . . . . .	71
5.14	Measured torques to ankles angles relation . . . . .	72
5.15	Push-Recovery experiment performance . . . . .	73
5.16	Push-recovery experiment forwards . . . . .	74
5.17	Push-recovery experiment forwards and backwards . . . . .	74



# List of Abbreviations and Symbols

<b>CoG</b>	Centre of Gravity
<b>CoM</b>	Centre of Mass
<b>Dof</b>	Degrees of Freedom
<b>F-T</b>	Force-Torque
<b>FRI</b>	Foot Rotation Indicator
<b>FZMP</b>	Fictitious Zero-Moment Point
<b>LIPM</b>	Linear Inverted Pendulum Model
<b>LQR</b>	Linear Quadratic Regulator
<b>TEO</b>	Task Environment Operator
<b>YARP</b>	Yet Another Robotic Platform
<b>ZMP</b>	Zero-Moment Point

$g$  gravity  $9.81m/s^2$

$l$  longitude

$m$  mass

$\tau$  torque

$\theta$  angular position

$\dot{\theta}$  angular velocity

$\ddot{\theta}$  angular acceleration

$x$  linear position

$\dot{x}$  linear velocity

$\ddot{x}$  linear acceleration

# Acknowledgments

Y de nuevo me hallo ante una página en blanco que llenar. Y de nuevo sucede que no sé por dónde empezar. Pero voy a intentarlo a ver qué tal sale.

En primer lugar quiero agradecer a mis directores de Trabajo Fin de Máster, Santiago Martínez de la Casa y Juan Miguel García Haro. Santi, gracias por darme la oportunidad de realizar este proyecto, por tus consejos y ayuda durante este año. Juanmi, gracias por animarme cada día que algo salía mal, por ayudarme a planificarme y a no pensar que ésto era imposible.

También quiero darles las gracias a mis padres por haberme animado a llegar hasta aquí y por toda su dedicación y sabios consejos. Gracias también al resto de mi familia y amigos, por preguntar en qué consistía mi trabajo y, de mejor o peor manera, intentar entender lo que os contaba.

Sobre todo, gracias a Jose, por soportar todos mis berrinches en épocas difíciles y por no dejarme jamás tirar la toalla.

Y por último gracias a todos los miembros del Robotics Lab que en algún momento me habéis echado una mano. Gracias a Raúl por sus tutoriales de git, a Pedro por sus cafés y a Juan H., Olaya, Elisabeth, Raúl y David por estar ahí cuando he necesitado vuestra ayuda.

A todos, muchísimas gracias.



# Resumen

El trabajo realizado en esta Tesis de Máster contribuye al desarrollo de la arquitectura de control de humanoides de cuerpo completo. Específicamente, este trabajo está centrado en el control de estabilidad de robots en presencia de perturbaciones externas. Todos los experimentos han sido realizados en el robot humanoide TEO (*Task Environment Operator*) del grupo de investigación RoboticsLab, en el departamento de Ingeniería de Sistemas y Automática de la Universidad Carlos III de Madrid.

Los robots humanoides representan el estado del arte en sistemas robóticos complejos. Entonces se requieren controladores robustos que puedan soportar perturbaciones desconocidas para una interacción segura con el entorno que rodea al robot. El análisis y la predicción del comportamiento de los humanoides es difícil debido al elevado número de grados de libertad y a la complejidad dinámica del sistema. Esa es la razón por la cual, generalmente, aproximaciones a modelos simples son utilizadas para simplificar el diseño y la dinámica. Sin embargo, las aproximaciones a modelos no siempre son como el sistema real. Las vibraciones causadas por la elasticidad estructural, el efecto de las pérdidas mecánicas, desviaciones en los sensores, irregularidades en el terreno, aproximaciones matemáticas y otros errores sistemáticos no tienen efectos positivos. Este trabajo demuestra cómo esos errores pueden ser tenidos en consideración

para proporcionar una mejora del modelo del robot.

En este trabajo, un control de estabilidad basado en el modelo ha sido analizado e implementado. Los modelos simples definen distintas estrategias de recuperación cuando aparecen perturbaciones inesperadas. Entonces, las tareas de *Push-Recovery* pueden ser clasificadas dependiendo de la intensidad de la perturbación. Esta tesis está centrada en la recuperación del sistema usando los tobillos del robot para perturbaciones de baja intensidad. El modelo lineal de péndulo invertido es utilizado para representar el cuerpo del robot cuando los tobillos son el punto de pivote del péndulo. Basándose en este modelo, la arquitectura del estabilizador ha sido desarrollada para mantener siempre el robot en una postura erguida.

El sistema de control desarrollado, al igual que lo hace el ser humano, procesa información procedente de varias fuentes sensoriales. El robot humanoide TEO está dotado con sensores de Fuerza-Par acoplados en las articulaciones de los tobillos. Estos sensores permiten obtener las fuerzas y los pares en las tres dimensiones. Esta información de realimentación da un importante indicador de estabilidad: el *Zero Moment Point* (*ZMP*). El *ZMP* indica cómo está el humanoide desplazado de su estado estable y natural.

Tanto el control de estabilidad basado en el modelo, como el procesamiento de la información sensorial han sido analizados a través de simulaciones y tests experimentales.

# Abstract

This Master Thesis contributes to the development of control architecture for whole-body humanoid robots. Specifically, this work is focused on humanoid robots stability control in presence of external disturbances. All the experimental tests are performed on the humanoid robot TEO (*Task Environment Operator*) from the RoboticsLab research group, in the Systems Engineering and Automation Department from the University Carlos III of Madrid.

Humanoid robots represent the state of the art in complex robot systems. Robust controllers that can handle unknown disturbances are required for a safely interaction with the surrounding environment of the robot. The analysis and prediction of humanoids behaviour is difficult due to the high number of degrees of freedom and the dynamic complexity of the system. That is the reason why, generally, simple model approaches are used to simplify the design and the dynamics. However, model approaches are not always as the real system. Vibrations caused by structural compliance, the effect of mechanical loosenesses, sensor deviations, irregularities in the ground, mathematical approaches and other systematic errors have no positive effects. This work demonstrates how this errors can be taken into account to provide an improvement of the robot model.

The stability model-based control has been analysed for controlling humanoid

robots. Simple models define distinct recovery strategies when unexpected perturbations appear. Then Push-Recovery tasks can be classified depending on the intensity of the disturbance. This thesis is focused on the system recovery using the robot's ankles for low intensity perturbations. Then the Linear Inverted Pendulum model is used to represent the robot body when the ankles are the pivot point of the pendulum. Based on this model, a stabilizer architecture has been developed in order to always maintain the robot in an upright posture.

The control system developed in this work, as well as human being does, processes information coming from sensorial sources. The humanoid robot TEO is provided with Force-Torque sensors located at its ankle joints. These sensors allow to obtain different measures of forces and torques in the three dimensional space. This feedback information gives an important stability indicator: the Zero Moment Point (ZMP). The ZMP indicates how the robot body is displaced from its natural and stable posture.

The performance of the stability model-based control as well the sensorial information processing have been tested through simulations and experimental trials.

Chapter **1**

# Introduction

## 1.1 Motivation and origin of the Thesis

Industry was one of the first fields of application of robotics, where the environment is mainly static, the tasks to be performed are repetitive and automated, and human interaction is quite low. For that reason, the idea of designing robots able to work in dynamic environments, with a high variety of tasks and interacting with humans and their environment, was fulfilled thanks to the evolution of new techniques related to robotics. Humanoid robots, physically similar to the human being, meet all that needs. Mainly, the possibility of moving, solves the problem of industrial robots that can only work in fixed areas. Moreover, the provision of artificial intelligence, allows the robot to interact with the surrounding environment in a more natural way, as a human being.

Nevertheless, the possibility of moving brings the problem of stability. Maintaining the humanoid robot in an upright posture and walking is a complex task related to control. For humans, walking is simple and we do it almost unconsciously, so we are not aware of its complexity. It has to be ensured that the robot is in an upright posture in order to not to fall over while it simultaneously is performing a set of movements previously defined to walk. Additionally, if

a disturbance appears and leads to an unbalanced situation, humans, unconsciously, try to stabilize moving their own body or the other limbs and the same behaviour is expected for a humanoid robot.

The idea behind this work is to ensure that the humanoid robot can balance itself in the presence of different disturbances. In a real environment, disturbances can origin from several sources. Since obstacle collisions and human interaction to irregular grounds which disturbs the robot balance. This thesis is focused on pushing disturbances that can lead the robot to an unbalanced situation and how the robot recovers from it.

As humanoid robots can work in different environment scenarios made for humans, then imitate human movements is the best chance to adapt them to our world.

## **1.2 Objectives**

This Master Thesis deals with the balance control of the humanoid robot TEO using Force-Torque sensors and attempts to discuss problems and issues that should be considered when the control system of a humanoid robot is designed.

The principle objectives of this work are:

- Provide a study of past and current research in the field of humanoid robots.
- Study basic concepts of humanoids control system. This means the study of general definitions, requirements and basic existing control architectures.
- Obtain information from the sensorial system of the real platform. This information will provide the necessary feedback to the control system.
- Propose a control architecture taking into account the functionality of the robot, its technical specifications and its design.

- Implement the results in the real platform. In order to check the applicability of the designed control architecture it is necessary to implement it in a physical platform. The platform used in this Master Thesis is the humanoid robot TEO from the Robotics Lab of University Carlos III of Madrid.



# Chapter 2

## Literature review

### 2.1 Trends in humanoid robotics

The word “Robot” first appeared in Karel Capek’s 1921 play *Rossum’s Universal Robots* where the *robots* were human-like machines made to replace human workers. It comes from the Czech word “Robota” which means “labour doing compulsory manual works without receiving any remuneration” or “to make things manually”. Robots are now very widely used in the manufacturing sector. Robotic technology has been developed and refined so successfully that an entire manufacturing process can be handled by robots alone.

The International standard ISO 3873 defines “Robot” as: “An automatically controlled, reprogrammable, multi-purpose, manipulator, programmable in three or more axes, which may be either fixed in place or mobile for use in industrial automation applications”. This definition restricts the area to only one type of robot, the industrial manipulator. But the inclusion of the perception of the environment and a capacity for action with some level of autonomy the robot leaves the manufacturing plant. The continuous evolution of robots needs a more general definition to include other types of robots in the global robotics area. The Oxford dictionary defines “Robot” as “a machine resembling a human being

and able to replicate certain human movements and functions automatically". Nowadays, the robot is leaving factories and laboratories and slowly entering society in the form of a service robot.

### 2.1.1 General classification of robots

The development of robotics through the ages, makes necessary to do a classification. Based on their ability to make different types of motion, their control architectures differ radically. Five groups of robotic systems can be distinguished by their motion control architecture: industrial, mobile, zoomorphic, anthropomorphic and hybrid robots.

*Industrial robots.* The main characteristic of this group is that all robots are stationary. Industrial robots, including industrial manipulators (Figure 2.1), are usually designed taking into account different requirements for velocity, load capacity, accessibility, etc., and always have a limited number of degrees of freedom. These robots are structured to move their end effectors in a determined working environment in one or several systems of coordinates. Although, exceptions may exist when a robot is guided in space (with moving platform) in order to perform a task in another environment. These robots are used when it is necessary to attend rather extensive but permanent working zones, working mainly with different types of objects and environments and does not exist human-robot interaction.

*Mobile robots.* This group has more motion capacity due to implemented wheel based platform systems (Figure 2.2). They can execute different tele-controlled tasks or are driven by the environmental information received from the integrated sensorial system. From the beginning of the sixties mobile robots were designed and implemented within industry. These robots were able to transport parts from one point of the production line to another, guided by pre-planned paths materialized by the electromagnetic or photoelectric bands from circuits mounted into the floor. From the beginning of the seventies a lot of work



**Figure 2.1:** ABB industrial manipulator

was related to major autonomy of mobile robots, like providing the mobile robot a vision system. Years later, when more complex and precise sensorial systems appeared, the development of control architectures for mobile robots was focused on the superficial intelligence and decision making systems (Salichs & Malfaz, 2012). Mobile robots provided with this kind of control system are usually able to plan motions and avoid obstacles. Today, research is also centred on human-robot interaction (Gonzalez-Pacheco et al., 2011).

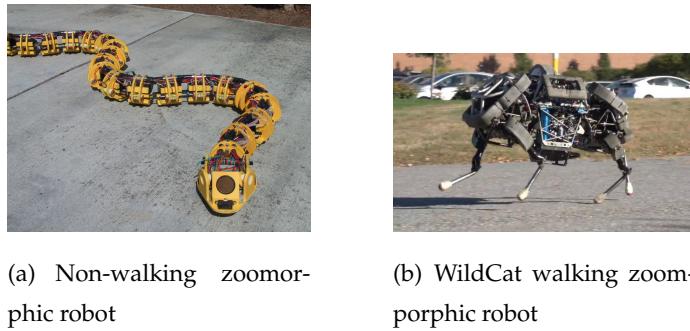


(a) TurtleBot mobile robot      (b) MAGGIE social robot from UC3M

**Figure 2.2:** Mobile robots

*Zoomorphic robots.* This type of robot is characterized by the locomotion system which imitates the locomotion of diverse living beings. Although there can

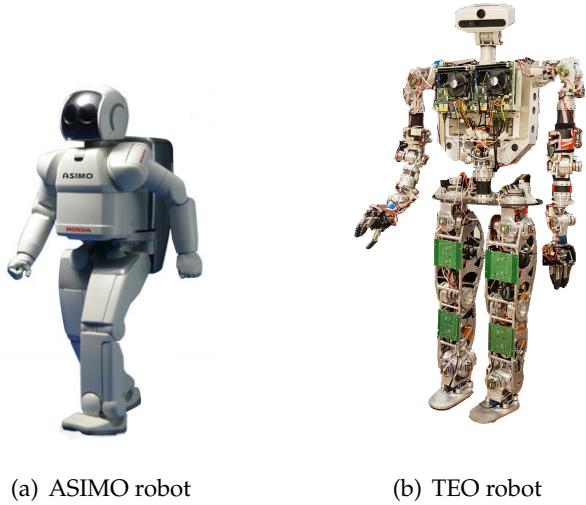
be a lot of morphological differences between all variations of zoomorphic systems, it is possible to distinguish two basic categories: walking and non-walking zoomorphic architectures. An example of non-walking zoomorphic robot is the modular snake-like robot in Figure 2.3 (a). Walking zoomorphic robots are developed to work in every kind of terrain and they have a really wide range of applications. Animal-like robots try to imitate the movements of animals and are usually constructed for research and entertainment. The control of this kind of robots is more complicated than control of a mobile or industrial robot because of the need to maintain the equilibrium at every stage of motion. Figure 2.3 (b) shows the WildCat walking zoomorphic robot developed by Boston Dynamics.



**Figure 2.3:** *Zoomorphic robots*

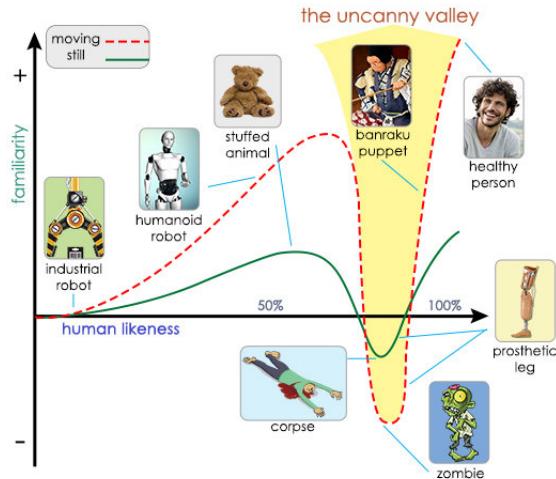
*Anthropomorphic robots (Androids)* or humanoid (bipedal) robots. These robots try to reproduce the body and behaviour of a human being. Presently the research on humanoids is increasing rapidly, although, there still remains a lot of work ahead. One of the basic challenges in this field is to reproduce human-like motion abilities beginning with the bipedal locomotion. The motion control architecture in this case is the most complex compared with the other robot types presented above. The main challenge is being able to control and coordinate in real time the dynamics of the whole body and maintain the equilibrium in the single support phase, i.e. when the robot is supported only by one foot. The

control architecture of this kind of robot is an aim of this research work and will be discussed and developed further in the following chapters. Figure 2.4 shows two examples of humanoid robots: (a) is Asimo Robot developed by Honda and (b) is TEO robot from UC3M.



**Figure 2.4:** *Humanoid robots*

Another complex aspect related to androids is the ability to reproduce the human upper body, especially the face. The difference between a humanoid robot and android is only skin-deep. The latter looks exactly like a human on the outside, but internally has the mechanics of a humanoid robot. But the human-like appearance can be controversial. In 1970, Masahiro Mori (Mori et al., 2012) presented his hypothesis about the *Uncanny Valley* (Figure 2.5). Mori's insight was that people would react with revulsion to human-like robots, whose appearance resembled, but did not quite replicate, that of a real human. The Uncanny Valley has become more relevant in the past few years since robots that actually look and move like humans are starting to become a reality. In fact, researchers currently debate over whether they should try to overcome the uncanny valley or simply design robots that are more mechanical in appearance.



**Figure 2.5:** *Uncanny valley*

*Hybrid robots.* These type of robots combine properties of various types of other robots. Usually they are a combination of a wheelbase (mobile robot) with an anthropomorphic body. Some examples are Justin robot from the German Aerospace Center (DLR) and TIAGO robot from PAL Robotics (Figure 2.6). They are both mainly involved in manipulation tasks (grasping, picking and placing, etc.) and the problem of locomotion is not considered as in humanoids.



(a) Justin robot (b) TIAGO robot

**Figure 2.6:** Hybrid robots

### 2.1.2 Humanoid robots

The history of the development of humanoid robots is curiously long and detailed. Leonardo da Vinci designed and possibly built the first humanoid robot (Figure 2.7). The robot, completely mechanical, was designed to sit up, wave arms and move its head via a flexible neck while opening and closing its jaw.

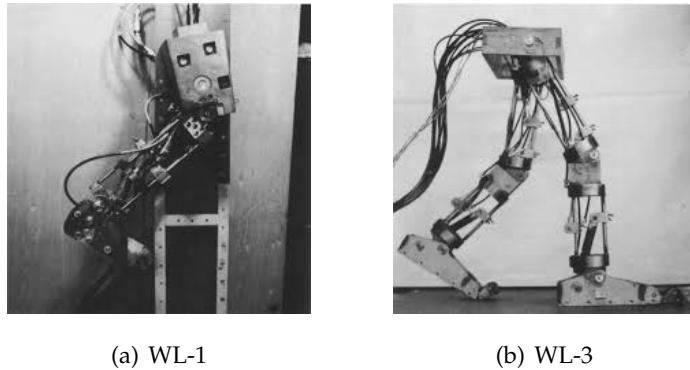


**Figure 2.7:** Leonardo da Vinci's mechanical knight

Other designs, related to the appearance of steam power and electricity appeared in the 19<sup>th</sup> Century. They were simple automates that imitated some human movements. The first bipedal machine was constructed by George Moore in 1983 but the real progress was achieved only at the end of the 1960s when the electronics, mechanics and materials became sufficiently advance to create such a complex system as a bipedal robot.

In the middle of the 1960s, the Japan University of Waseda launched different research projects lead by Kato and as a result, the series of WL robots appeared. They were only lower body robots (Figure 2.8).

But the real first works about bipedal robots with humanoid appearance (complete torso) were carried out about 1970 by authors Kato (Kajita et al., 2014) and Vukobratović (Vukobratovic et al., 1970). The first anthropomorphic robot WABOT-1 (Figure 2.9 (a)), was exhibited by Kato in 1973. Using a very simple control diagram, the robot was able to perform a few slow gaits, maintaining its balance at all times. This achievement, was the first one that encouraged



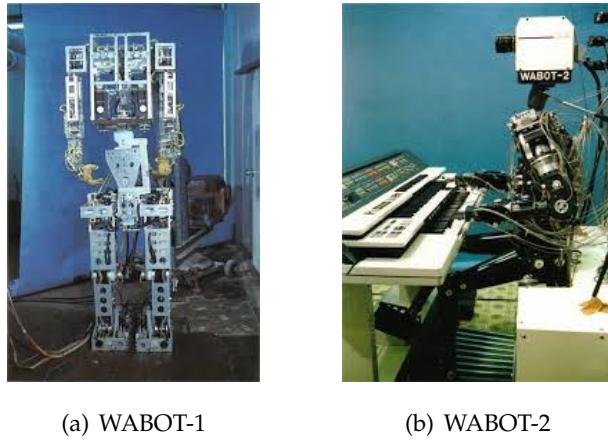
**Figure 2.8:** University of Waseda WL robots

researchers about humanoid robots and their locomotion.

At the same time, Vukobratović and his research team were studying stability in biped systems in the former Yugoslavia, basing on a new stability criterion, presented in 1972, as *Zero-Moment Point* (ZMP) (Vukobratović & Borovac, 2004). Taking into account the dynamic effects produced during a walking, from then until now, the ZMP stability criterion has been the most used in humanoid or biped robotics.

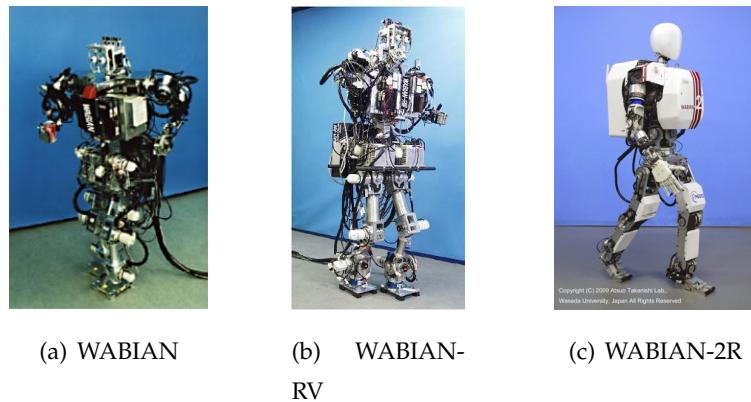
The second prototype WABOT-2 of Waseda University was developed between 1980 and 1984. The robot musician WABOT-2 can converse with a person, read a normal musical score with its eyes and play tunes on an electronic piano. The WABOT-2 can be mentioned as one of the first personal robots.

Years later, the project continued in the Japanese university and the WABIAN series (Figure 2.10) were constructed as improvements of the previous ones. These robots had advances in their degrees of freedom, better mechanical structure, better stability and bigger weight of objects to be transported. The WABIAN series is still in development with its latest prototype WABIAN 2-R. New technologies and materials make a huge difference between this and the first versions. An important advantage of its mechanical design is the use of 2-DOF in the waist which enables more fluent walking motions.



(a) WABOT-1

(b) WABOT-2

**Figure 2.9:** *WABOT series*

(a) WABIAN

(b) WABIAN-RV

(c) WABIAN-2R

**Figure 2.10:** *WABIAN series*

The development of humanoid robotics was not only at universities also in industry. The Honda Motor Company was one of the main pathfinder industries involved in robotics. In the mid 1980s, they began creating biped walking robots (only lower body parts). But the rise of humanoid robots started with the development of P1 robot in 1993 (Kajita et al., 2014). The project began in secret years before, after the exhibition of WABOT-2 playing the piano. The next version, P2 in 1996 (180 centimetres high and 210 kg weight), was the first humanoid able

to walk in a stable enough way and carry its processor and battery on its back. After that, robots P3 and ASIMO were its advanced versions, reducing height and weight of the robot.

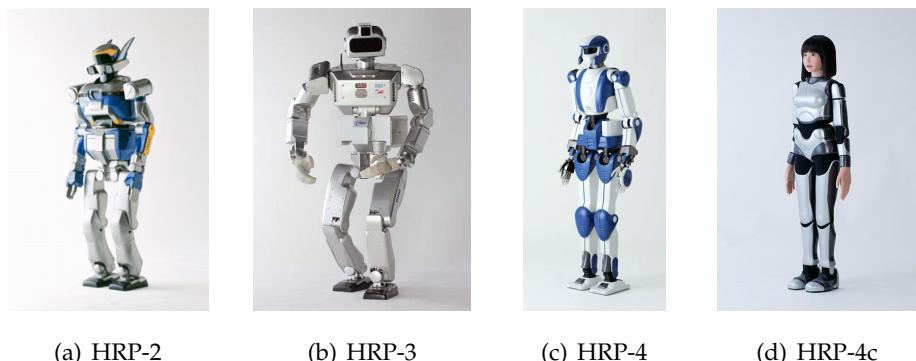


**Figure 2.11:** Honda humanoid prototypes. From left to right: E0 (1986), E1(1987), E3(1989), E5(1992), P1(1993), P2(1996), P3(1997), ASIMO(2000)

Undoubtedly, ASIMO is the culmination of two decades of research in humanoid robotics by Honda. ASIMO (130 centimetres and 54 kg) is able to run, walk (at 2.7 km/h) on uneven and slope surfaces, climb stairs, grasp objects among other tasks. It also can avoid moving obstacles as it moves through its environment.

Other of the most advanced humanoid robots is the HRP-2 . The Humanoid Robotics Project (HRP) is led by the National Institute of Advanced Industrial Science and Technology (AIST) and Kawada Industries in Japan (Kaneko et al., 2004). The main outcome of this project was HRP-2 which is taller (140 cm) but lighter (58 kg) than Honda's Asimo. HRP-2 was the first finished prototype in the HRP series to demonstrate walking (2 km/h maximum speed) on uneven surfaces, getting up from a fallen position and to interact with humans and help in domestic service tasks. Subsequently, HRP-3 was developed based on the HRP-2 experience with additional dust and splash-proof protection, enhanced hand coordination, improved cooling systems, and prolonged operation time. In

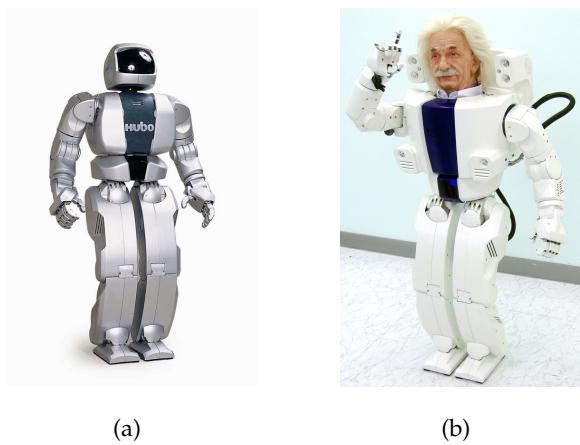
particular, the prototype of HRP-3 (Kanehiro et al., 2006) was developed using a distributed control system and a real-time Ethernet communication network. Later, HRP-4 has a new slim, lighter and athletic mechanical design (weight of 39 kg, height of 151 cm) to improve its functioning in the domestic environment. In addition to the mechanical upgrades, this robot has several key improvements which include LAN and Wireless LAN networks for internal and external communications. Finally, HRP-4c (Kaneko et al., 2011), also referred to as the cybernetic humanoid robot, is the most recently developed humanoid robot in the HRP series with a female appearance which is 158 cm tall and weighs 43 kg. HRP-4c has human like facial expressions. It should be noted that all the HRP series robots have rigid joints which are controlled by local PID controllers and an on-line ZMP based trajectory generation.



**Figure 2.12: HRP series**

Meanwhile, the Korean Advanced Institute of Science and Technology (KAIST) has been developing a series of walking robots since 2001 which consist of KHR-1, KHR-2 and KHR-3 where the mechatronic design was evolved over three versions. Hubo (KHR- 3) is the latest humanoid robot developed at KAIST in 2004 which has 41 DoF, stands 125 cm tall and weighs 55 kg as shown in Figure 2.13 (a). The aim of the Hubo project is to provide a reliable platform for implementing dynamic walking, navigation and image processing algorithms (Park

et al., 2005). In 2008, Hubo was integrated with an android head resembling Albert Einstein. The head used RC servo motors for facial expressions and the robot was called Albert Hubo (Park et al., 2008) (Figure 2.13 (b)). In late 2009, a running experiment was reported on Hubo (Cho et al., 2009) with the maximum speed of 3.24 km/h. The tracking control system which used in Hubo humanoid robot is based on independent PD position feedback loops running at 1 kHz which is the traditional approach for trajectory tracking in robotics.



**Figure 2.13:** (a) HUBO robot (b) Albert HUBO robot

The fastest walking robot is Petman by Boston Dynamics with a human like walking speed of 4.4 miles per hour (7 km/h approx.) (*Boston Dynamics Website*, 2016). The aim of the Petman project is to study the feasibility of chemical testing using a fully articulated robotic mannequin for the US military. Petman weighs 80 kg and stands about 1.75 m tall and has hydraulic actuators which in addition to high power output, provide a degree of compliance to absorb the ground impacts. Currently the robot can walk, go up the stairs, squat and kneel. Recently, Boston Dynamics has received more funding from DARPA (Defense Advanced Research Projects Agency in USA) to develop a more advanced version of Petman which is called Atlas.

Going to small sized humanoid robots, iCub (from the Instituto Italiano di

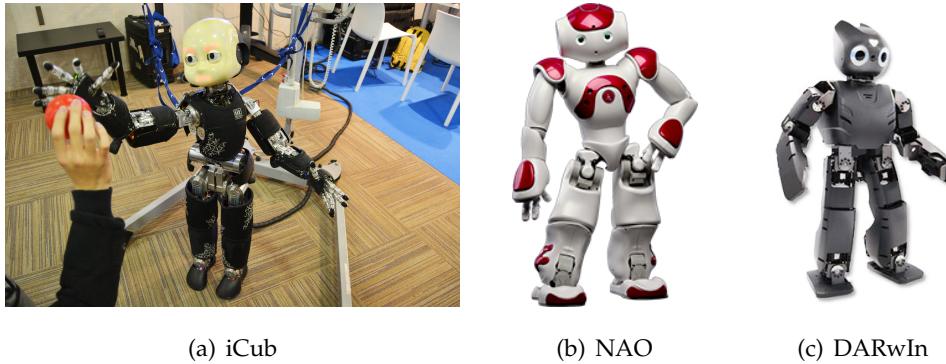


**Figure 2.14:** Boston Dynamics humanoid robots

Tecnologia) is a humanoid robot that was developed in five years by the RobotCub (*The RobotCub Project Website*, 2016) consortium to provide the cognitive science research community with an open source platform to study cognition. In other words, iCub is used as an open source robotic platform to study how a human child learns the basic motor skills as well as learning and recognizing different objects. iCub is a full humanoid robot with a head, arms, hands, waist, and legs that are actuated by 53 motors. It is the size of a 3.5 year old child with a compact, modular, mechatronic architecture, stands 104 cm tall, and weighs less than 23 kg. Its internal communication network is based on CAN, which connects the local DSP joint controllers to the central computer located at its head.

Other open-source small sized humanoids (Figure 2.15) are NAO humanoid robot (58 cm, 4.3 kg and 21 to 15 DoF) developed by Aldebaran Robotics in Paris, the Fujitsu HOAP-3 humanoid robot (60 cm and 8.8 kg), and the DARwIn-OP (54.5 cm, 2.9 kg, 11 DoF) developed by RoMeLa at Virginia Tech in collaboration with the University of Pennsylvania, Purdue University and Robotis Co.

In Spain, two main projects are being developed. On the one hand the company PAL Robotics develops humanoid robot REEM-C (Figure 2.16) which is 1.65 m tall and weighs 80 kg. This robot has grasping ability, object and faces



**Figure 2.15:** Mini humanoid robots

recognition, obstacle avoidance, human-robot interaction and speech and voice recognition in several languages. It is fully compatible with ROS (Robotic Operating System).



**Figure 2.16:** REEM-C robot from PAL Robotics

On the other hand, the Robotics Lab of the University Carlos III of Madrid launched the humanoid robot project, the Rh-0 in its first phase (2002-2004), the Rh-1 (2005-2007) and finally the Rh-2 (2007-Present). The Rh-1 humanoid robot (Figure 2.17) is 1.45 m tall, weighs 48.5 kg, has 21 DOFs, can walk at about 0.8 Km/h (Arbulu et al., 2005), recognizes faces, and responds to voice commands. The Rh-2 is an improvement of its previous version and it is 165 cm tall, weighs about 63 kg and it has 28 DoF. This platform is the one used in the experimental

trials of this work, so its technical characteristics will be seen in Section 3.



**Figure 2.17:** *Rh-1 humanoid robot*

## 2.2 Bipedal locomotion

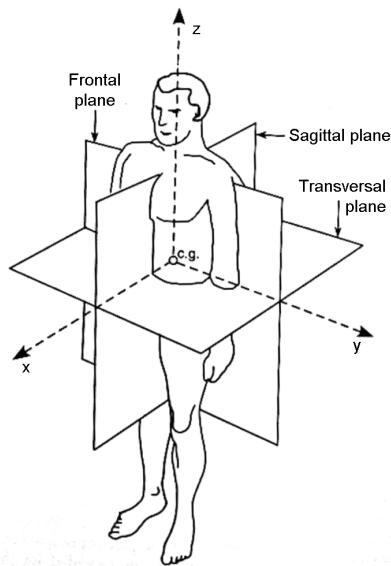
Artificial bipedal locomotion is a complex task and humans have the ideal locomotion. Therefore the best way to reproduce a type of motion for a walking machine is to copy human motion.

Human walking is an automated motion, carried out even unconsciously. This locomotion process is a repetitive execution until some perturbations are detected. In humans, the muscular system modifies forces acting during the walk in order to maintain balance. The study of human walking and the muscles involved in, brings very complex relationships and it requires some simplifications in anthropomorphic legged mechanisms in order to reduce complexity from the mechanical and control points of view.

The analysis of the human walking is fairly recent. McGeer (McGeer, 1990) built a passive walker in 1990 and showed that his two-legged walker could reproduce stable gait without any controls. However, the most progress was revealed in the active bipedal locomotion (Hirai et al., 1998), (Kaneko et al., 2004),

(Park et al., 2005). This type of locomotion is developed and implemented as artificial human-like bipedal motion based on the previous planning of each step and the real-time automatic control of its execution.

Generally, the study of the human body is based on three basic planes: sagittal, transversal and frontal (Figure 2.18). It is important to mention that the most important motions related to locomotion occur in the sagittal plane because it coincides with the main walking direction. However, the combination of joints of sagittal and frontal planes, give the stability of the locomotion cycle.



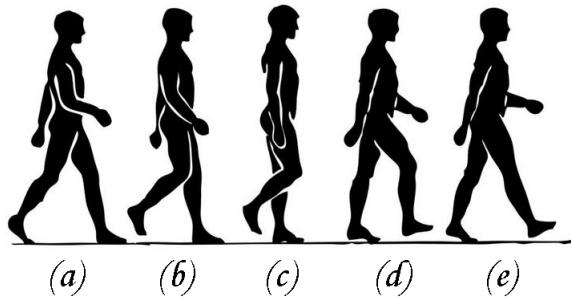
**Figure 2.18:** Axes division of human body.

The basic characteristic of bipedal locomotion is the permanent change of the situation when the mechanism is supported by one foot (single support phase) and when both feet are in contact with the ground (double support phase). The second situation is statically stable and there are no additional moments affecting the robot. In terms of balance, the first situation is statically unstable because when one foot is on the ground, the other is transferred from the back to the front position. It produces lateral accelerations affecting the mechanism and all

the weight remains in only one support foot. Each of these two cases present different dynamical situations and should be taken into account in artificial gait synthesis and control.

Robot walking, as humans, is performed in a three phase cycle (Figure 2.19). The cycle is divided in two, left and right steps. At the beginning, the human is in a stable position with both feet on the ground (Figure 2.19 (a)) and all the body weight is transferred from one foot to the other. Then the step generation starts when the right foot leaves the ground in the swinging phase (Figure 2.19 (b) - (d)). After the right foot touches the ground (Figure 2.19 (e)), the next (left) step with the same basic phases is started, and the whole cycle ends.

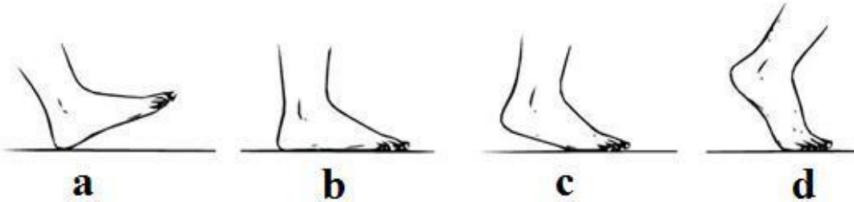
Swinging phase has also three sub-phases: acceleration, swinging and deceleration (Figure 2.19 (b), (c) and (d), respectively). The acceleration phase takes its name due to the acceleration of the lifting leg that stops being supported in the ground and gives the impulse to the step. Once the support leg is overtaken, the lifting leg starts the swinging phase in order to reach the ground with its consequent deceleration.



**Figure 2.19:** Half cycle phases of biped walking.

As in the case of the legs, the same occurs with the feet, what adds complexity to control the walking cycle. During a gait, a human foot has four different

phases as one can see in Figure 2.20. In (a) it is shown how the body weight is supported when the heel is touching the ground. In (b), the foot remains totally plane. In (c), the heel lifts and the weight goes to the front part of the foot. Finally, in (d) the foot is not in contact with the ground and it starts to swing.



**Figure 2.20:** Phases of foot support during a walk.

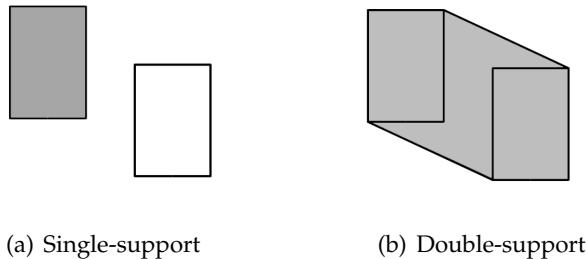
Almost all humanoid robots do not have articulated feet, due to the high complexity. They use plane and rigid feet and the control is done in the ankle joint. Some of them, as robot HRP-4c (Kaneko et al., 2011), have a joint called "active toe joint" which allows the movement of the toes. The reason of this improvement is related to reach a more natural and fluent walking.

However, in order to perform a stable walk, is not only necessary the lower body movement. The upper body is also involved in recovery movements. In an example, if a person stumbles. Unwittingly, he or she would move the opposite arm of the unbalanced leg, or even more, moving the torso in order to not to fall down. In the case of a biped robot, the same strategy is followed, which means a high increase of complexity in the robot stability control.

### 2.3 Biped balance/equilibrium

It is important for humanoid locomotion to avoid overturning during the walking or even to reach an upright position of its body. To prevent falling down, a necessary and sufficient condition is to ensure that there exists a contact area between the foot and the ground and not a line or a point (Vukobratovic et

al., 2007). Given a rectangular-shaped foot, the support area of the robot will be a polygon. In case that only one foot is touching the ground (single-support), the support area is the contact region between the sole and the ground, i.e., the footprint (Figure 2.3(a)). On the contrary, when both feet are touching the ground (double-support), the support area will be determined by the footprints and the common tangents between them (Figure 2.3(b)). It means that in double-support phase of the walk, the support area is bigger than in single-support phase, so stability is higher.



**Figure 2.21:** Support areas depending on the support type.

## 2.4 Zero Moment Point (ZMP)

In (Vukobratovic et al., 2007), Vukobratović makes a distinction between the term “balance” used in the sense of maintaining an upright posture, and “equilibrium”, taking into account the D’Alembert’s principle. The D’Alembert’s principle states that the resultant of the external forces and the kinetic reaction acting on a body equals zero (condition of kinetic equilibrium). When the humanoid is falling since it is rotating about one foot edge, the D’Alembert’s principle still holds for a point on the foot edge where the pressure force acts. Anyway, this case can not be contemplated as balanced in the sense of the definition previously provided. This point is called *Center of Pressure* (*CoP*) and it is known as the point, in a single-support phase, where the pressure forces (nor-

mal to the sole) are equivalent to a single resultant force exerted at the point where the resultant moment is zero.

From the concept of the CoP, appears a new term known as *Zero-Moment Point (ZMP)*. The ZMP is a point inside the support area where, always, the resulting dynamic reaction of the biped system is acting. In a more specific definition, the ZMP is a point inside the support area where the resultant of all forces and torques acting on the full body, is equal to zero.

Vukobratović explains the difference between the CoP and ZMP: CoP and ZMP coincide only when both are inside the support area. When the ZMP goes to the edge of the support area, the humanoid body loses balance and it will fall down. In that case, the ZMP has no sense existing even the CoP.

Goswami presented that, mathematically, was possible that the point could be outside of the support area and continue satisfying the equilibrium conditions (Goswami, 1999). This point, called *Foot Rotation Indicator (FRI)*, is defined as the point on the contact area between the ground and the foot, inside or outside the support area, where the resultant moment of the forces and torques applied on the foot are normal to the surface. Forces and torques applied mean the forces and torques at the ankle joint, and also other external forces, the foot weight and reaction forces between the foot and the ground.

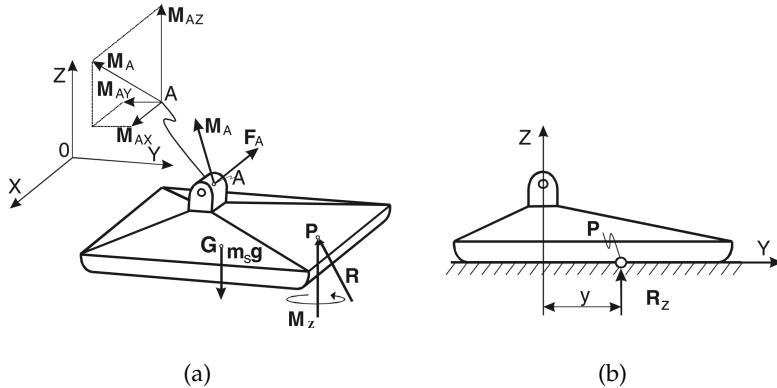
However, Vukobratović held that the ZMP can only exist inside the support area of the robot. When the ZMP comes close to the edge, any force or moment applied to the system, will produce a rotation about the foot edge and the robot will fall down. In this case, the reaction force of the ground will be at the foot edge and, therefore, it can not be considered as ZMP because there is no stability ensured. That is why the author suggests to denote the point as *Fictitious ZMP* o *FZMP*, if it is outside the support area.

When the robot walking is enough slow to consider almost static, appears the term *pseudo-ZMP*, which is the projection over the ground of the *Center of Gravity (CoG)* of the system. In such case, lateral accelerations are so small and

can be omitted and the *pseudo-ZMP* = ZMP. Although the *pseudo-ZMP* do not give precise information about the balance of the mechanism, it can be used in order to make a first approximation in control and design of a humanoid robot.

#### 2.4.1 Equations of ZMP

Let us consider the locomotion mechanism in the single-support phase, with the whole foot in contact with the ground (Figure 2.22 (a)). To simplify the analysis we can neglect the part of the mechanism above the ankle of the support foot (point A) and replace its influence by the force  $F_A$  and moment  $M_A$ , whereby the weight of the foot itself acts at its gravity center (point G). The foot also experiences the ground reaction at point P, whose action keeps the whole mechanism in equilibrium.



**Figure 2.22:** Forces acting on the foot of the bipedal mechanism (Vukobratović & Borovac, 2004)

In general, the total ground reaction consists of three components of the force  $R(R_x, R_y, R_z)$  and moment  $M(M_x, M_y, M_z)$  exerted at the foot-ground contact point. During the support phase, it is assumed there is no shifting in the contact point, which means that horizontal reaction force  $R_x$  and  $R_y$  balances the horizontal component of the force  $F_A$ , whereas the vertical reaction moment  $M_z$  represents the moment of friction reaction forces that balances the vertical com-

ponent of the moment  $M_A$  and the moment induced by the force  $F_A$ .

However, due to an unidirectional nature of the connection between the foot and the ground (it is obvious that the ground reaction force induced by foot action is always oriented upwards) horizontal components of all active moments ( $M_A$ ) can be compensated for only by changing position of the reaction force  $R$  within the support polygon. This is illustrated in Figure 2.22 (b) where a planar case in  $y - z$  plane is represented.

The moment  $M_{Ax}$  is balanced by shifting the acting point of the force  $R_z$ , whose intensity is determined from the equation of balance of all the forces acting on the foot, by the corresponding distance  $y$ . It is necessary to emphasize that all the time the reaction force is within the area covered by the foot, the increase in the ankle moment will be compensated for by changing the position of this force  $R_z$ , and no horizontal components of the moments  $M_x$  and  $M_z$  will exist. This is the reason why in Figure 2.22 at point  $P$  only the  $M_z$  component exists.

However, if the real support polygon is not large enough to encompass the appropriate position of the force  $R$  to balance the action of external moments, the force  $R$  will act at the foot edge and the uncompensated part of the horizontal component of the reaction moment will cause the mechanism rotation about the foot edge, which can result in the mechanism overturning. Therefore, it can be said that the necessary and sufficient condition for the locomotion mechanism to be in dynamic equilibrium is that for the point  $P$  on the sole where the ground reaction force is acting,

$$\begin{aligned} M_x &= 0, \\ M_y &= 0. \end{aligned} \tag{2.1}$$

That is why the *Zero-Moment Point* is called the contact point with the ground ( $P$ ) where there no exist shifting, i.e., moments  $M_x$  y  $M_y$  are zero.

From Figure 2.22, static equilibrium equations for the supporting foot are obtained:

$$\sum \vec{F} = 0 \Rightarrow \vec{R} + \vec{F}_A + m_s g = 0 \quad (2.2)$$

$$\sum \vec{M}_O = 0 \Rightarrow \vec{OP} \times \vec{R} + \vec{OG} \times m_s g + M_A + M_z + \vec{OA} \times F_A = 0, \quad (2.3)$$

where  $\vec{OP}$ ,  $\vec{OG}$  and  $\vec{OA}$  are radius vectors from the origin of the coordinate system  $O_{xyz}$  to the ground reaction force acting point ( $P$ ), foot mass center ( $G$ ), and ankle joint ( $A$ ), respectively, while the foot mass is  $m_s$ . If we place the origin of the coordinate system at the point  $P$  and project equation (2.5) onto the z-axis, then the vertical component of the ground reaction moment (actually, it is the ground friction moment) will be

$$M_z = M_{fr} = -M_A^Z + (\vec{OA} \times F_A)^Z \quad (2.4)$$

In a general case, this moment is different from zero and can be reduced to zero only by the appropriate dynamics of the overall mechanism. However, the projection of equation (2.5) onto the horizontal plane gives:

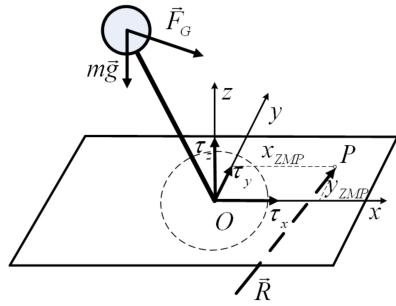
$$(\vec{OP} \times \vec{R})^H + \vec{OG} \times m_s g + (M_A)^H + (\vec{OA} \times F_A)^H = 0 \quad (2.5)$$

This equation is a basis for computing the position of the ground reaction force acting point ( $P$ ) which gives the ZMP position.

### 2.4.2 Relation between COG and ZMP

When a humanoid robot is in the single-support phase during a walking cycle, its dynamics can be represented by a 3D linear inverted pendulum where all body mass is concentrated in the CoG connected to the supporting foot point by means of a massless telescopic leg (Kajita et al., 2001). It allows to decouple controller design for the sagittal (x-z) and the lateral (y-z) motion.

Now, let us consider that the inverted pendulum instead of only one contact point considered above, has a contact polygon as the surface in contact with the ground (Figure 2.23).



**Figure 2.23:** 3D Linear Inverted Pendulum with a contact polygon (Kaynov, 2008)

Inertial  $\vec{F}_G$  and gravity  $m\vec{g}$  forces act on the point mass located in the CoG of the humanoid robot. The contact of the pendulum with the ground produces a reaction force  $\vec{R}$  and reaction moment  $\vec{M}_P$  at point P. For any other point of the support polygon (taking point 0 as origin), the moment  $M_0 = [\tau_x, \tau_y, \tau_z]^T$  produced by the ground reaction force  $\vec{R}$  is represented:

$$M_0 = M_P + \vec{OP} \times \vec{R} \quad (2.6)$$

If it is considered point P to be the ZMP of the system, then from the interpretation of the ZMP presented above  $M_P = 0$ . In this case we can denote vector

$\overrightarrow{OP} = [x_{ZMP}, y_{ZMP}, z_{ZMP}]^T$  and equation (2.6) gives the following equation:

$$\begin{pmatrix} \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} x_{ZMP} \\ y_{ZMP} \\ z_{ZMP} \end{pmatrix} \times \overrightarrow{R} \quad (2.7)$$

From the other side, applying Newton's law of mechanics to system in Figure 2.23:

$$m\overrightarrow{a_G} = \overrightarrow{R} - m\overrightarrow{g} \quad (2.8)$$

where  $\overrightarrow{a_G} = [\ddot{x}, \ddot{y}, \ddot{z}]^T$  is the acceleration of the CoG. From equation (2.8) it is obtained:

$$\overrightarrow{R} = m \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} + g \end{pmatrix} \quad (2.9)$$

Substituting equation (2.9) into the equation of balance of moments (2.7) it is obtained:

$$\begin{pmatrix} \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = m \begin{pmatrix} x_{ZMP} \\ y_{ZMP} \\ z_{ZMP} \end{pmatrix} \times \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} + g \end{pmatrix} \quad (2.10)$$

After a cross product and taking into account that  $z_{zmp} = 0$ , because the ZMP lies into the ground:

$$\begin{pmatrix} \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = m \begin{pmatrix} y_{ZMP}(\ddot{z} + g) \\ -x_{ZMP}(\ddot{z} + g) \\ x_{ZMP}\ddot{y} - y_{ZMP}\ddot{x} \end{pmatrix} \quad (2.11)$$

From (2.11), it can be stated the ZMP position of the system:

$$x_{ZMP} = -\frac{\tau_y}{m(\ddot{z} + g)} \quad (2.12)$$

$$y_{ZMP} = \frac{\tau_x}{m(\ddot{z} + g)} \quad (2.13)$$

If it is supposed that the CoG always remains within the horizontal plain intersecting the  $z$  axis in the point  $z_c$  (one of the constraints of the 3D-LIPM model), then the vertical component of the CoG acceleration  $\ddot{z} = 0$ . Then, finally, equations (2.12) and (2.13) are:

$$x_{ZMP} = -\frac{\tau_y}{mg} \quad (2.14)$$

$$y_{ZMP} = \frac{\tau_x}{mg} \quad (2.15)$$

When the ankle joint is displaced from the ground as in Figure 2.24, ZMP equations take the form:

$$x_{ZMP} = -\frac{\tau_y + hF_x}{mg} \quad (2.16)$$

$$y_{ZMP} = \frac{\tau_x + hF_y}{mg} \quad (2.17)$$

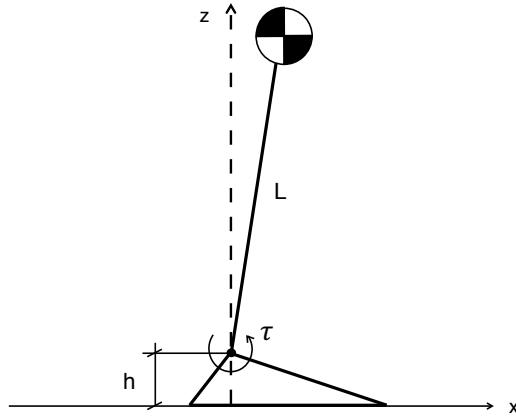
where  $h$  is the distance between the ground and the measuring point, i.e., the height of the sole. These are the *ZMP equations*, without forgetting it was considered before that lateral accelerations of the COG  $\ddot{x} = 0$  and  $\ddot{y} = 0$ . One can see that moments  $\tau_x$  and  $\tau_y$  in  $x$  and  $y$  directions respectively affect the ZMP position of the mechanism and it can loose balance because of their change.

If we take into account these lateral accelerations of the COG, the ZMP can be expressed as a function of the acceleration of the CoG as:

$$x_{ZMP} = x_{COG} - \frac{z_c}{g} \ddot{x}_{COG} \quad (2.18)$$

$$y_{ZMP} = y_{COG} - \frac{z_c}{g} \ddot{y}_{COG} \quad (2.19)$$

These equations can only be applied to compute ZMP in single-support phase. In the case of the double-support phase, it is necessary to calculate the weighted average of the sensor measurements form both feet as recommended in (Kajita



**Figure 2.24:** Single inverted pendulum model

et al., 2014, pp. 82-83). Therefore, the resulting equations for ZMP in double-support phase are:

$$x_{ZMP} = -\frac{x_{ZMP}^R \cdot F_z^R + x_{ZMP}^L \cdot F_z^L}{F_z^R + F_z^L} \quad (2.20)$$

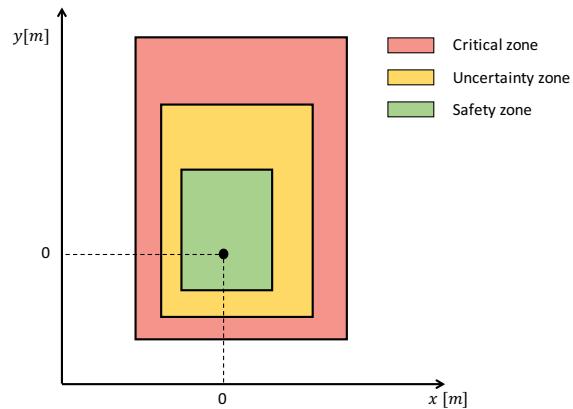
$$y_{ZMP} = \frac{y_{ZMP}^R \cdot F_z^R + y_{ZMP}^L \cdot F_z^L}{F_z^R + F_z^L} \quad (2.21)$$

where the upper index *R* represents the right foot and *L* the left one.

#### 2.4.3 ZMP areas.

As mentioned before, ZMP is a point in the sole which depends on the forces and moments applied to the robot. Therefore, depending on the magnitude of those forces, ZMP will change and it becomes a dynamic parameter. As suggested in (Vukobratovic et al., 2007), three regions are defined depending on the position of the ZMP as one can see in Figure 2.25. In the balanced area (safe region), the control action will not actuate. In the nearly critical region, the control action will actuate as a secondary solution. This may be the case of a walking task. As humans do, the robot may use its arms in order to reduce the zero-moment point

position closer to the safe region. Finally, in the critical region, the stabilizer will actually disconnect the ongoing task and actuate on the full body. Even if this region is still stable, the balance may be easily lost.



**Figure 2.25:** ZMP stability regions in single-support.

# Chapter 3

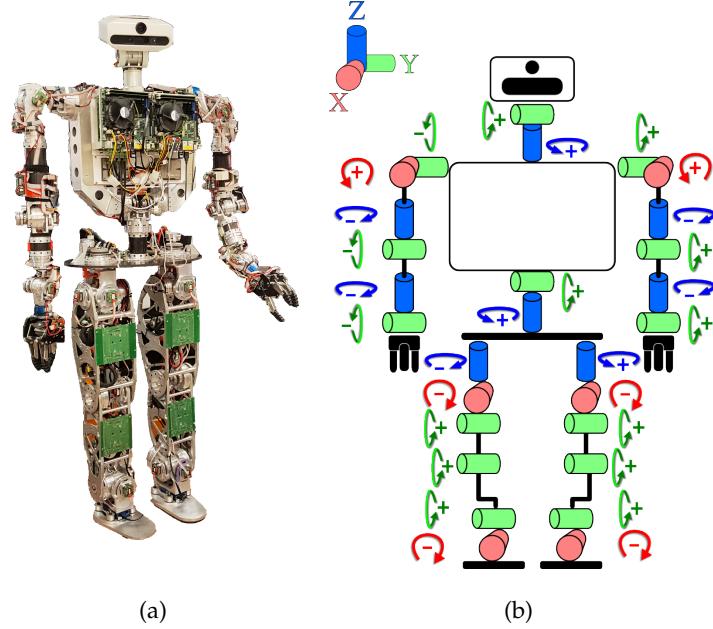
## Platform description

### 3.1 Humanoid robot TEO

Humanoid robot RH-2, also known as TEO (*Task Environment Operator*), from University Carlos III of Madrid, is an advanced version of humanoid robots RH-0 and RH-1. TEO is 165 cm high overtaking 150 cm of RH-1 and 120 cm of RH-0. It weighs about 63 kg and it can carry about 2 kg of payload. It has 26 DOFs (28 DOFs taking into account head motors), more DOFs than in previous versions. In Figure 3.1 (b) one can see the robot DOFs, besides their direction of rotation, being 6 DOFs for each leg, 6 DOFs for each arm, 2 DOFs for the torso and 2 DOFs for the head.

The robot has 4 microprocessors: locomotion, manipulation, artificial vision tasks and last, the main processor which manages the others. The locomotion processor, that controls the legs and the torso, will be responsible for getting the sensors information and maintain the robot in a balance and upright position, being static or in a walking cycle. The manipulation processor controls the movement of the arms and the head. The processor responsible for the computer vision uses a camera with infrared sensor ASUS located in the head.

The communication system is based on the CAN-bus protocol. Making a



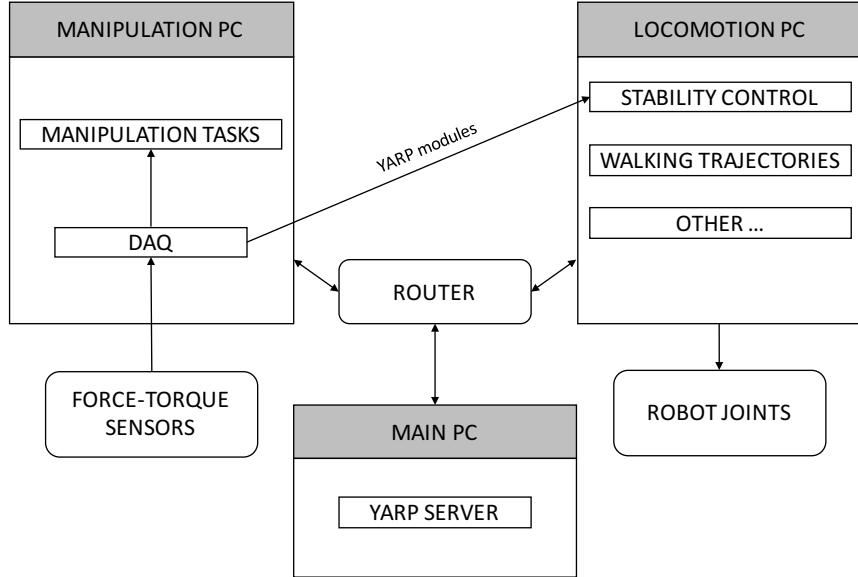
**Figure 3.1:** (a) Robot TEO and (b) joint distribution diagram

sagittal and transversal division, there are 4 CAN-bus lines: 1 line per each arm and neck, and 1 line per each leg and torso.

For data acquisition, the robot has a inertial sensor located at the trunk and Force-Torque sensors located at the robot ankles and wrists. F-T sensors are plugged into real time data acquisition PCI cards. Including F-T sensors is an important difference and advantage respect RH-2 predecessors. They allow to close the control loop and then, to obtain a kind of feedback which is necessary to accomplish tasks successfully.

The Control level is divided into 2 layers. At joint level, each servo not only closes the servo loop, it also synchronizes with other devices and absolute encoders provide a feedback of joint motors. At task-oriented control level, high level executions are done. This layer is directly connected with the joint level control. Examples of appliance are manipulation of objects or stability control which is the aim of this project. Figure 3.2 shows the general communication

diagram.



**Figure 3.2:** Basic communications diagram

## 3.2 Force/Torque sensors

Force-Torque (F-T) sensors are based on strain gauge sensors arranged in such a way that allows to obtain force and moment measures in all axes of the 3D space.

The sensors used in the platform, are the commercial JR3 F-T sensors described in Table 3.1. Look at the full scales difference between the sensors used in the wrists joints and the ones used in the ankle joints. Ankle sensors must be able to support greater forces and moments including the ones exerted by the own robot.

According to the manufacturer, the two first digits of the model show the

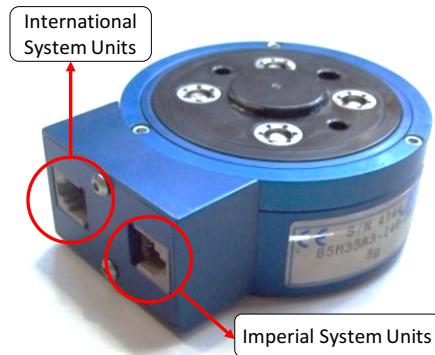
Joint	Model	$F_{x,y}$	$F_z$	$M_{x,y,z}$
Wrist	50M31A	100N	200N	5 Nm
Ankle	85M35A	250N	500N	212Nm

**Table 3.1:** F-T sensor models and characteristics. [JR3 Inc.]

sensor diameter, followed by the serie, and the next two digits, the thickness. As mentioned before, the ankle sensors are bigger and they support greater forces and moments. The sensors used in this work are the ones mentioned in Table 3.1 for ankle joints.

M series sensors include inner electronics in order to filter noise, digital output to use a data acquisition PCI card from the same manufacturer and an analogical output option. The nominal precision of all sensors of M series is 1% of full scale, and a 1/4000 full scale resolution.

The sensors used in this work, provide the option of acquiring data in International System Units or Imperial System Units, according to Figure 3.3.



**Figure 3.3:** JR3 model 85M35A Force-Torque sensor

### 3.3 Data Acquisition

The PCI cards used for data acquisition are PCI 1592D from JR3 Inc, which has 4 ports (named as in Figure 3.4). The sensors are plugged through a 6 or 8 pinout cables (RJ-11 and RJ-45, respectively). In the case of RJ-45, two pinouts are not used. The PCI card uses these cables to receive high speed data and provide power supply to the sensors. About the PCI supply, it is provided by the PCI slot form the computer where it is installed.



**Figure 3.4:** JR3 data acquisition card

In order to access to received data from the sensors, it is necessary to access to card memory, specifying the memory address for each available data. These addresses can be found at Appendix A.

It is important to take into account that the forces and torques obtained from

the sensors and processed by the card, are in the International System Units. Forces are given in Newton [N] and torques are given in tenths of a newton per meter [ $\text{dN}\cdot\text{m}$ ].

### 3.3.1 Acquisition program

The *jr3pci4channelYarp* program (available in <https://github.com/lpinel/LoliRepo>) reads data from the 4 F-T sensors of the robot. Data read from the Data Acquisition Card is scaled to SI units, clustered to a YARP Bottle object and sent through YARP ports. Figure 3.5 summarizes the data acquisition procedure and Figure 3.6 shows a screenshot of the running program.

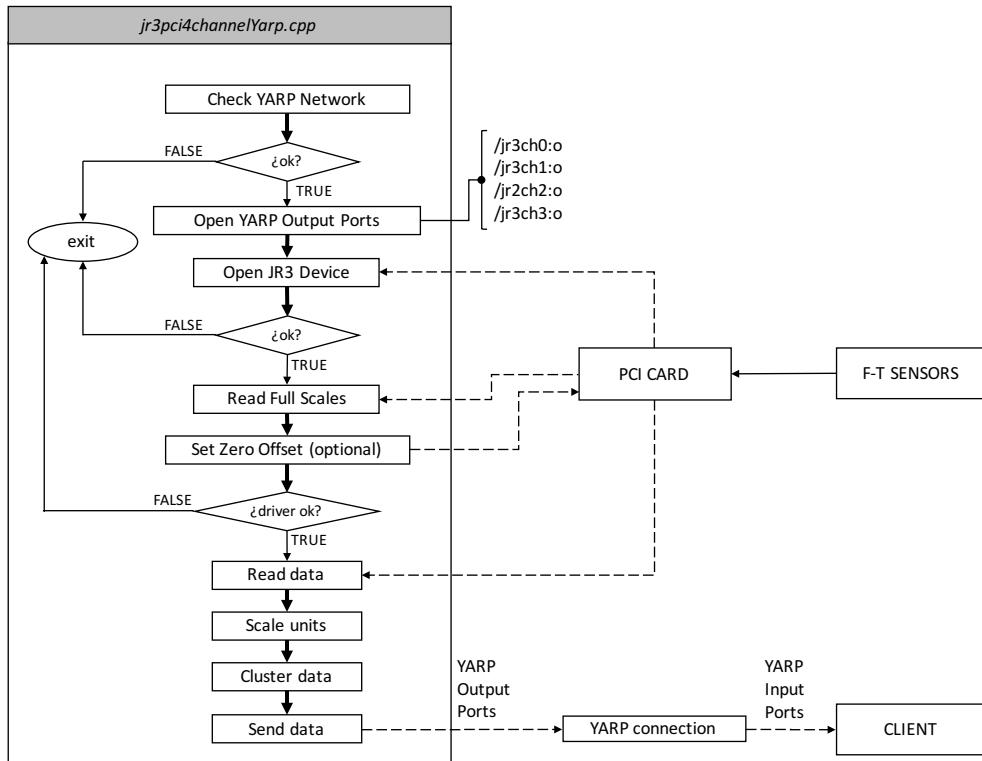
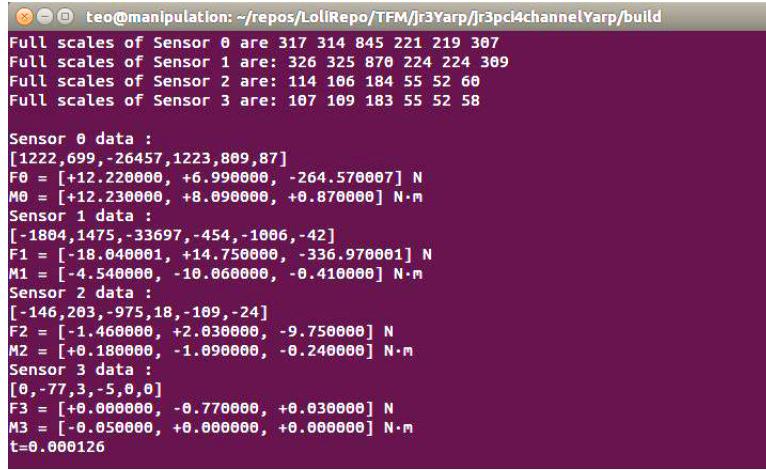


Figure 3.5: Data acquisition diagram



```

teo@manipulation: ~/repos/LoliRepo/TFM/Jr3Yarp/Jr3pcl4channelYarp/build
Full scales of Sensor 0 are 317 314 845 221 219 307
Full scales of Sensor 1 are: 326 325 870 224 224 309
Full scales of Sensor 2 are: 114 106 184 55 52 60
Full scales of Sensor 3 are: 107 109 183 55 52 58

Sensor 0 data :
[1222,699,-26457,1223,809,87]
F0 = [+12.220000, +6.990000, -264.570007] N
M0 = [+12.230000, +8.090000, +0.870000] N·m
Sensor 1 data :
[-1804,1475,-33697,-454,-1066,-42]
F1 = [-18.040001, +14.750000, -336.970001] N
M1 = [-4.546000, -10.060000, -0.410000] N·m
Sensor 2 data :
[-146,203,-975,18,-109,-24]
F2 = [-1.460000, +2.030000, -9.750000] N
M2 = [+0.180000, -1.090000, -0.240000] N·m
Sensor 3 data :
[0,-77,3,-5,0,0]
F3 = [+0.000000, -0.770000, +0.030000] N
M3 = [-0.050000, +0.000000, +0.000000] N·m
t=0.000126

```

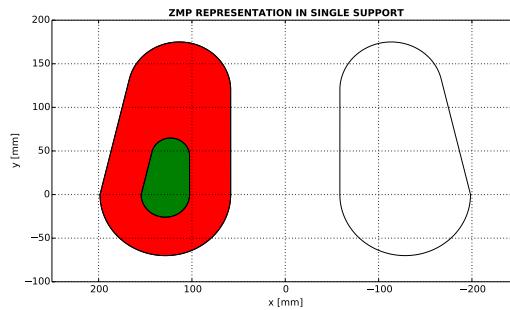
**Figure 3.6:** Screenshot of the data acquisition interface

Program cycle time is about  $20\mu s$  for each sensor, thus four sensors reading cycle time is about  $80\mu s$ . The problem comes when the data is sent through YARP ports and there is a client receiving and processing this data. Then the update rate hugely decreases between  $10 - 50ms$ , depending on the reader processing cycle. That occurs because the arrival of updates is delayed until the client completes processing and no updates will ever be lost on the client side (Metta et al., 2006).

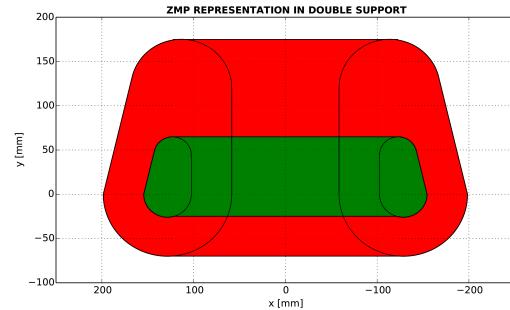
Using these Force-Torque measurements from the sensors located at the ankles of the robot, the ZMP values are obtained using equations (2.14) and (2.15), where the distance between the ground and the sensor center is 25.5 millimetres (sensor height is obtained from Appendix B).

For a better visual observation of the ZMP data, it has been developed a Python User Interface using the module *Matplotlib* to represent the ZMP in the ground X-Y plane with the sole borders in both single and double support. The ZMP stability areas are also represented according to the limits obtained later in section 5.6 as shown in Figure 3.7. As the experiments have been done in the sagittal plane, the limits obtained are:

$$-0.02m \leq x_{ZMP} \leq 0.065m \quad (3.1)$$



(a)



(b)

**Figure 3.7:** Python interface for Single and Double Support with delimited ZMP areas

# Chapter 4

## Control Architecture

### 4.1 Introduction

Vukobratovic (Vukobratovic et al., 1970) was one of the first researchers involved in the stability of bipedal robots, followed by Kajita (Kajita et al., 2001) and Kim(Kim & Oh, 2004). In all their studies, the biped robot was usually represented by a planar inverted pendulum with the base representing the foot and the ankle joint. Latest research is focused on divide robot balance control into different strategies: ankle, hip and step strategies. The basis of the three strategies are close to the ZMP areas explained in previous sections. When the robot is in a stable posture and a disturbance is applied, depending on the magnitude or the that disturbance, the robot will react different. If the change of the ZMP position remains in a stable area, the control will react by the motion of the ankle joints to recover the robot balance. Nevertheless, if the perturbation increases and the ZMP position reaches an uncertain-stability area, it will be also necessary to move the hip joints in addition to the ankles to recover balance. Even a gait will be necessary if the loss of stability is unavoidable.

A humanoid is an electromechanical system, so it should have all type of errors: structure flexion, small blacklashes between motion parts, etc. Also it will

operate in a co-existing environment with humans, so the disturbances are unexpected at any time. Therefore, the Stabilizer is an essential element to provide stable human-like walking of a humanoid robot. The Stabilizer should perform two basic operations:

1. When the humanoid robot walks, it should correct the robot's walking trajectory in order to provide the secure position at any time of its motion.
2. When the humanoid robot has stopped, it should control its posture.

Thus, the Stabilizer can be decoupled into ZMP and Attitude controllers. Attitude controller is applied when velocity and accelerations of the robot boy are measured. This Master Thesis will deal with the issue of maintaining an upright posture while the robot is in a static position (without following a motion pattern) by only moving the robot ankle joints. Then the Stabilizer designed in this work, will be responsible to control the robot ZMP position using the F-T sensors located at the robot ankles.

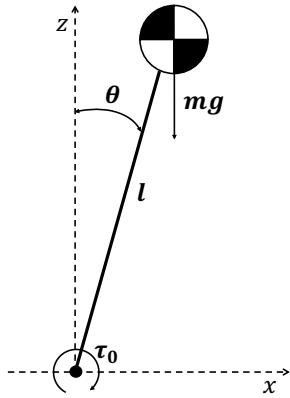
## 4.2 Inverted Pendulum Model

The inverted pendulum is the most basic model used to simplify humanoids' body. The basis of the pendulum are a mass  $m$  linked to a pivot point by means of a massless link of longitude  $l$  as in Figure 4.1.

The mass  $m$  represents the total mass of the modelled system, a humanoid robot in this case, located at its Centre of Mass (CoM), and the longitude  $l$  is the distance between the pivot point to the CoM. Its dynamical model in a planar, for example XZ case, is expressed by the equation (4.1), if gravitational force is considered the only force acting in the system.

$$\tau_0 = -ml^2\ddot{\theta} + mgl \sin \theta \quad (4.1)$$

where  $\tau_0$  is the torque generated by the ankle joint,  $\theta$  its angular position,  $\ddot{\theta}$  its angular acceleration and  $l$ , the distance between the joint and the CoM. For

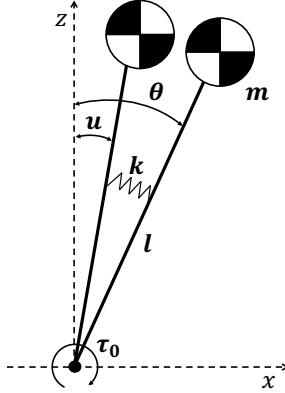


**Figure 4.1:** Single inverted pendulum.

simplification of a control task, let us make a linearisation of nonlinear differential equations, taking the approximation that perturbations are small enough to consider  $\sin \theta = \theta$ . It is not defined how small these angles have to be in practice to apply the linearisation assumptions, but in this case it is assumed that  $\theta \leq 10^\circ$ . Then, equation (4.1) changes to linearised equation (4.2)

$$\tau_0 = -ml^2\ddot{\theta} + mgl\theta \quad (4.2)$$

The main complexity of this model is the fact that equation (4.2) does not give the possibility of controlling the ZMP by angular position of the ankle joint. To overcome this problem, the inverted pendulum model can be slightly modified. The link of the pendulum which connects the ankle joint to the concentrated mass (CoM) is generally assumed to be rigid. However, in the real humanoid mechanism it is slightly flexible because the leg length is relatively long and the mechanical structure suffers from flexibility and small backlashes. Because of this compliance, the humanoid robot exhibits the characteristics of a lightly damped structure (Kim & Oh, 2004). For example, in a static case when the ankle joint is under position control, a pushing external force can easily excite an oscillation. This oscillation exists even when the position error in every joint is zero. This phenomenon is prevalent in the fast dynamical gait; therefore it is



**Figure 4.2:** Single inverted pendulum with compliant joint.

very important to implement a control mechanism allowing ZMP fast correction considering the stiffness of the humanoid robot links. The most suitable model in this case will be a single mass inverted pendulum with compliant joint as shown in Figure 4.2, where  $u$  denotes the ankle joint reference angle and  $\theta$  denotes the actual inclined angle produced by the compliance of the mechanical structure of the humanoid,  $k$  denotes the stiffness of the leg and  $\tau_0$  is the torque produced by the motor of the ankle joint to place the inverted pendulum into the desired angular position. Then, the torque  $\tau_0$  should be expressed as:

$$\tau_0 = k(\theta - u) \quad (4.3)$$

Taking the Laplace transform of equation 4.2, it is obtained:

$$T(s) = -mgl\theta(s) + ml^2 s^2 \theta(s) \quad (4.4)$$

The Laplace transform of equation (4.3) is:

$$T(s) = k(\theta(s) - U(s)) \quad (4.5)$$

Clearing  $\theta(s)$  from equation (4.5) and placing it into the equation (4.4) and

simplifying, the transfer function is obtained:

$$\frac{T(s)}{U(s)} = k \frac{-s^2 + (\beta - \alpha)}{s^2 + \alpha} \quad (4.6)$$

where:

$$\alpha = \frac{k - mgl}{ml^2} \quad (4.7)$$

$$\beta = \frac{k}{ml^2} \quad (4.8)$$

On the other hand, from equation (2.14) relating the moment produced by the ground reaction force around  $y$  axis with  $x$  ZMP direction (the planar XZ case of the inverted pendulum is considered) we can get:

$$\tau_y = -mgx_{ZMP} = -F_z x_{ZMP} \quad (4.9)$$

and then the Laplace transform of the equation (4.9) is:

$$\tau_y(s) = -F_z x_{ZMP}(s) \quad (4.10)$$

For the static equilibrium of the system, the moment generated by the motor of the ankle joint should compensate the moment produced by the ground reaction force:

$$\tau_0 = \tau_y \quad (4.11)$$

The relation between  $\tau_y$  and  $x_{ZMP}$  is linear, therefore, placing (4.10) into (4.6) we get the following transfer function relating ZMP to ankle joint position:

$$\frac{x_{ZMP}(s)}{U(s)} = -k_1 \frac{-s^2 + (\beta - \alpha)}{s^2 + \alpha} \quad (4.12)$$

where  $k_1 = \frac{k}{mg}$ .

In equation (4.12)  $x_{ZMP}(s)$  is the output and  $U(s)$  is the input of the system.

The state space representation of the dynamical system in the standard form is:

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} \quad (4.13)$$

$$y = \mathbf{Cx} + Du \quad (4.14)$$

where  $\mathbf{x}$  is a state ( $n$ -vector),  $y$  is the output (escalar),  $u$  - control (scalar),  $\mathbf{A}$  -  $n \times n$  constant matrix,  $\mathbf{B}$  -  $n \times 1$  constant matrix,  $\mathbf{C}$  -  $1 \times n$  constant matrix and  $D$  a scalar.

To obtain the state representation of the inverted pendulum system let us define state variables  $x_1$  and  $x_2$  by:

$$x_1 = \theta \quad (4.15)$$

$$x_2 = \dot{\theta} \quad (4.16)$$

where angle  $\theta$  denotes the rotation of the pendulum about the ankle and  $\dot{\theta}$  its angular velocity. We consider the ZMP as the output of the system, then  $y = x_{ZMP}$  in the XZ planar case. From the definition of state space equations (4.13) - (4.16) and the linearised equations of the inverted pendulum motions (4.2) and (4.3) we obtain the state space representation of the system in the *Controllable Canonical Form*:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\alpha & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \quad (4.17)$$

$$y = \begin{bmatrix} -k_1\beta & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} k_1 \end{bmatrix} u \quad (4.18)$$

### 4.3 Feedback in state space. The Linear Quadratic Regulator

The quadratic optimal control method is one of the control methods applied in state space systems and it provides a systematic way of computing the state feedback control gain matrix (Ogata, 2010). Given the state space system equation

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (4.19)$$

the LQR determines the matrix  $\mathbf{K}$  of the optimal control vector

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) \quad (4.20)$$

so as to minimize the performance index

$$J = \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt \quad (4.21)$$

where  $\mathbf{Q}$  is a positive-definite (or positive-semidefinite) Hermitian or real symmetric matrix and  $\mathbf{R}$  is a positive-definite Hermitian or real symmetric matrix. Note that matrices  $\mathbf{Q}$  and  $\mathbf{R}$  determine the relative importance of the error and the expenditure of the energy of the control signals. The linear control law given by equation (4.20) is the optimal control law. Therefore, if the unknown elements of the matrix  $\mathbf{K}$  are determined so as to minimize the performance index, then  $\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t)$  is optimal for any initial state  $x(0)$ .

The optimum  $K$  matrix is obtained from equations (4.22) and (4.23).

$$K = R^{-1} B^T P \quad (\text{continuous case}) \quad (4.22)$$

$$K = (R + B^T P B)^{-1} B^T P A \quad (\text{discrete case}) \quad (4.23)$$

where  $P$  is a positive-definite Hermitian or real symmetric matrix obtained from the algebraic Riccati Equation:

$$P \rightarrow A^T P + P A - P B R^{-1} B^T P + Q = 0 \quad (\text{continuous case}) \quad (4.24)$$

$$P \rightarrow A^T P A + P - A^T P B (R + B^T P B)^{-1} B^T P A + Q = 0 \quad (\text{discrete case}) \quad (4.25)$$

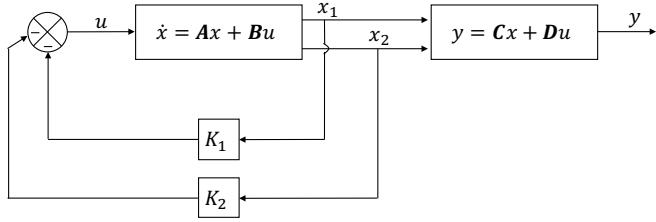
In order to obtain the controller design for further simulations and experiments, the following mechanical parameters of the inverted pendulum (corresponding to Rh-2 humanoid robot) were taken:  $m = 62.589$  kg,  $l=0.8927$  m,  $k=200$ . The high stiffness value is due to the rigidity of the pendulum (the leg in this case). If it has a high stiffness, the pendulum will behave as a so rigid link, but if it is lower, the pendulum will be considered as a flexible link and will react in a slower way.

For the optimum response of the control system, we take  $Q = C^T C = \begin{bmatrix} 0.9487 & 0 \\ 0 & 0 \end{bmatrix}$  and  $R = Q$ . After the LQR controller was designed, the following parameters were obtained using a sample time  $T = 0.03$  s.

$$\mathbf{A} = \begin{bmatrix} 1.003 & 0.03003 \\ 0.2096 & 1.003 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0.0004502 \\ 0.03003 \end{bmatrix}; \\ \mathbf{C} = \begin{bmatrix} -1.3060 & 0 \end{bmatrix}; \quad D = 0.3257; \quad (4.26)$$

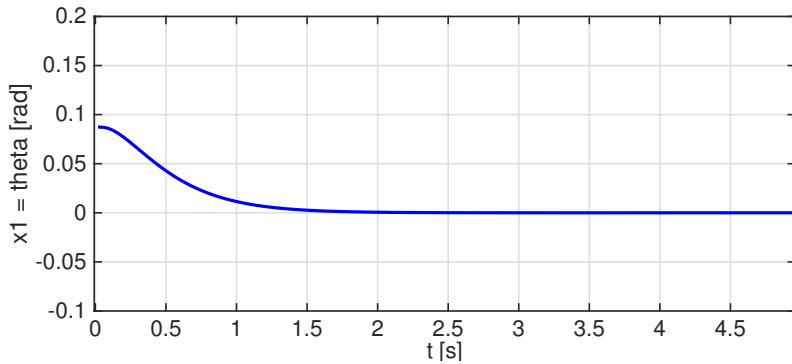
$$\mathbf{K} = \begin{bmatrix} 13.5366 & 5.1035 \end{bmatrix} \quad (4.27)$$

The block diagram showing the optimal configuration for the single inverted pendulum system is presented in Figure 4.3. The controller maintain desired ( $x_{ZMP}$ ) position , and also  $\theta$ , of the single inverted pendulum close to zero. Thus, the reference input of the control system in Figure 4.3 is zero. A further point of interest for the humanoid robot is to have command tracking so that the real humanoid robot joints could be positioned anywhere and this can be achieved by adding an offset to the desired angle of the ankle joint.



**Figure 4.3:** LQR controller block diagram

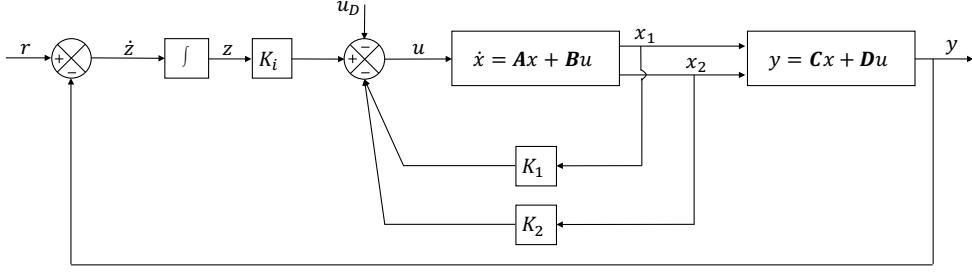
Figure 4.4 shows simulation results with the designed LQR control system when the initial pendulum angle  $\theta(0) = 5^\circ \simeq 0.08\text{rad}$ . It can be seen how the inverted pendulum system returns to its reference position (zero).



**Figure 4.4:** Simulation of inverted pendulum with initial conditions.  $\theta(0) = 0.08\text{rad}$

The state space representation (4.17), (4.18) is a controllable canonical form that is important for the LQR controller design. It is desired to keep the actual ZMP, measured and computed by force-torque sensors located in the feet of the humanoid robot, close to its stable reference position as was discussed in previous sections. As the system is a type 0 plant, it is necessary to insert an integrator in order to design a ZMP servo control system (type 1) and remove the steady state error. Therefore, we feed the output signal  $y$  (which indicates the real ZMP)

back to the input and an integrator in the feedforward loop as is shown in Figure 4.5. Here,  $z$  denotes the error between the actual and the reference ZMP,  $u$  represents the commanded angle to the system and  $u_D$  is the corresponding angle to the reference ZMP ( $r$ ).



**Figure 4.5:** ZMP LQR control system.

Thus, referring equations (4.17) and (4.18) and Figure 4.5 and considering the actual ZMP position as the output of the system and  $r$  as the reference input signal we obtain the equations for the closed loop system as follows:

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} \quad (4.28)$$

$$y = \mathbf{Cx} + \mathbf{Du} \quad (4.29)$$

$$u = -\mathbf{Kx} + K_i z - K_u u_D \quad (4.30)$$

$$\dot{z} = r - y = r - (\mathbf{Cx} + \mathbf{Du}) \quad (4.31)$$

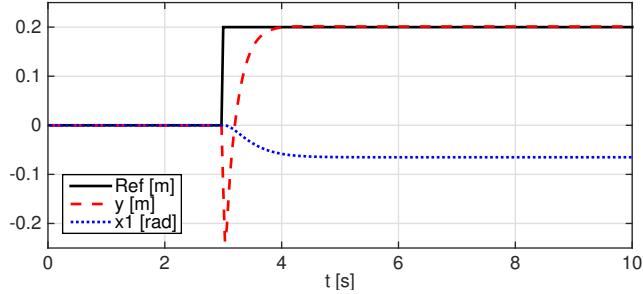
For the type 1 servo system, the state error equation is given by:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ z \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ 0 \end{bmatrix} u \quad (4.32)$$

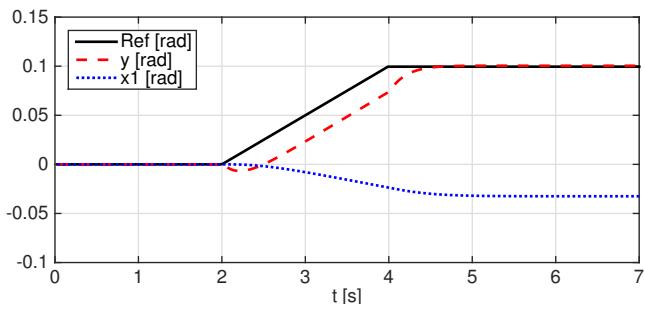
and the control signal  $u$  is given by:

$$u = \begin{bmatrix} -\mathbf{K} & K_i \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ z \end{bmatrix} - K_u u_D \quad (4.33)$$

Figure 4.6 shows simulation results with the designed LQR control system with an integrator in the direct control loop. One can see how the output  $y$  reaches the step reference of ZMP. Note that the output goes to negative values when the reference suddenly changes its value. This abrupt change makes the output derivative to reach higher values, so it can be solved reducing the abrupt change of the reference signal, i.e., using a ramp rise instead of an abrupt step. In Figure 4.7 the reference change is smaller and the output is smoother.



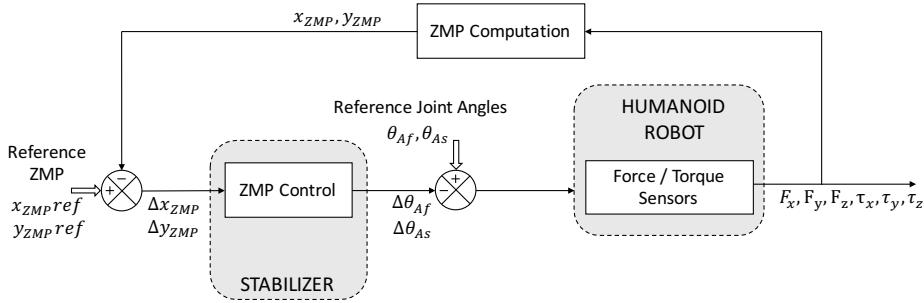
**Figure 4.6:** Simulation of linear inverted pendulum step response.



**Figure 4.7:** Simulation of linear inverted pendulum smooth step response.

## 4.4 Stabilizer

Previously it was shown that the dynamics of a humanoid robot can be a single inverted pendulum and how the pendulum maintains a desired position thanks to the controller designed. Now, let us introduce the detailed stabilizer structure (Figure 4.8).



**Figure 4.8:** Stabilizer architecture.

The sensorial system of the robot consisting of two six-axis force-torque sensors located at the robot ankles, provide the controller the real distribution of the forces and torques  $F_x, F_y, F_z, \tau_x, \tau_y, \tau_z$  at the contact point of the foot with the ground. After the actual ZMP position  $x_{ZMP}, y_{ZMP}$  is computed, the ZMP errors ( $\Delta x_{ZMP}, \Delta y_{ZMP}$ ) can be estimated. These errors are the input data for the Stabilizer and it controls the error in ZMP positioning of the humanoid robot by the motion of the ankle joints ( $\theta_{As}$  and  $\theta_{Af}$  are sagittal and frontal ankles angular positions).

## 4.5 Control strategies

Humans are capable of performing numerous dynamical movements in a wide variety of complex and novel environments while robustly rejecting a large spectrum of disturbances. Human movements such as a forward step and rapid arm rotations allow them to maintain overall balance in non-stability situations. Many researchers have studied how humans unwittingly use their body parts to recover balance as a response of external disturbances and make an approach for studying the stability of humanoid robots.

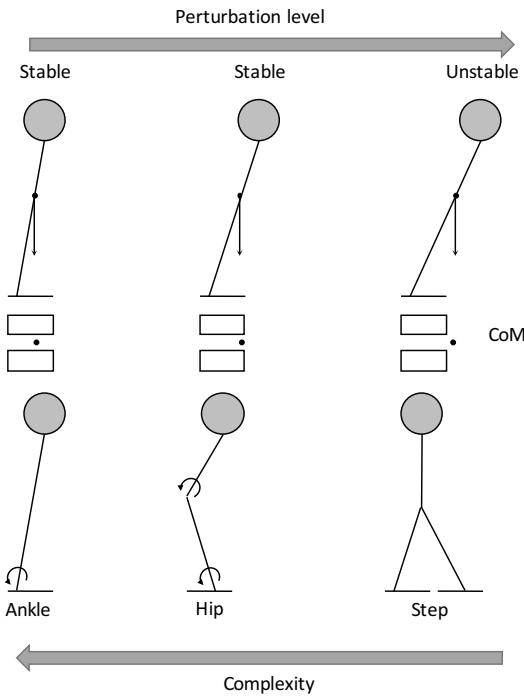
When the humanoid robot is in a stable posture, perturbations may appear and they can be classified according the direction of action of that perturbations. All of them can be decomposed in anteroposterior disturbances (sagittal plane) and mediolateral disturbances (frontal plane). Studies of quiet stance have suggested separate postural strategies for balance in both planes depending on the stance position (Winter et al., 1996). There are three main mechanisms that can be applied to regain balance in such planes depending on the level of the disturbance: ankle, hip and step strategies (Nenchev & Nishio, 2007).

The first is the ankle strategy. This strategy is applied in the sagittal plane or anteroposterior disturbances. For low intensity disturbances, the body can be considered as a nearly stiff pendulum, and balance adjustments are mainly made in the ankle joint, with the body balancing like a single inverted pendulum.

In the hip strategy, the resulting motion is mainly applied to the hip joints. It can be applied independently or in combination of the ankle strategy. The hip joint movement is triggered when the external disturbance increases and the ankle strategy is not enough to keep balance. The hip strategy, same as ankle one, acts in the anteroposterior direction.

The last one is the step strategy. When these postural corrections become insufficient, the base of support must be adjusted. The modification of the support base leads new balance stability limits.

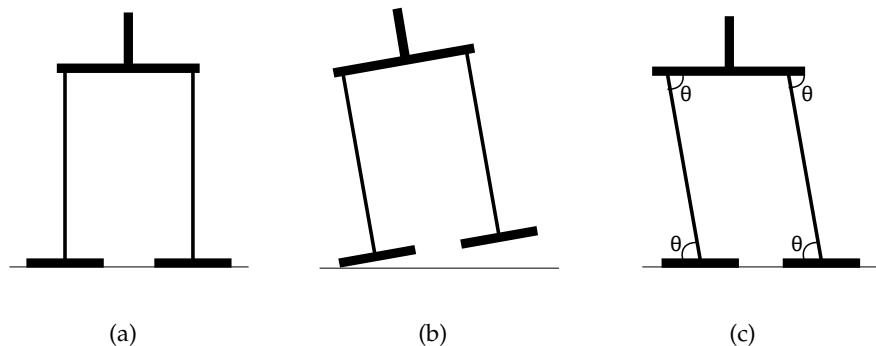
Figure 4.9 summarizes these three strategies and shows the levels for strategy triggering. These limits are not fixed and they depend of the humanoid design as the sole surface or the height of the whole body, environmental conditions, i.e., standing in a flat surface has different strategy limits than in a narrow surface.



**Figure 4.9: Recovery strategies.**

In the mediolateral direction or frontal plane, the disturbances are compensated by the lateral movement of the hip joint in the case of upright stance. Double support in frontal plane, means there are two support points and a pendulum can not be considered. Both legs and the trunk of the robot make a parallelogram with the ground (see Figure 4.10 (a)). If a disturbance appears, and the hips maintain their perpendicular angles to the body, the feet will begin to lose contact with the ground as shown in Figure 4.10 (b). Then, to maintain stability, the angles of the parallelogram must keep on their relationship without losing

contact between feet and the ground, and the motion is applied to ankle and hip joints (Figure 4.10 (c)).



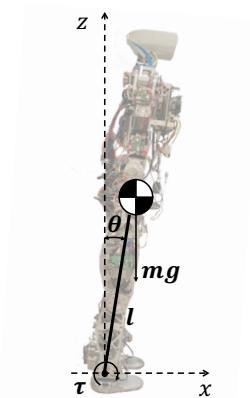
**Figure 4.10:** Influence of hip and ankle angles in the frontal plane stability.



# Chapter 5

## Experimental results

In this chapter the experimental results are presented and discussed. For all experiments it has been used the humanoid robot TEO. The aim of this trials is to show the feasibility of the model-based stabilizer designed in section 4.4. Only the ankle strategy is considered in the sagittal plane when the robot is in double support phase (Figure 5.1). Double support phase in the sagittal plane can be represented as a single inverted pendulum as both ankles have the same movement along the  $x$  axis. All the experimental trials have been performed in a flat ground controlled environment.



**Figure 5.1:** Experimental setup of robot TEO

## 5.1 Ankles angle - ZMP relation

First, in order to experimentally obtain the ankles angular position and ZMP measurement relation we did several tests. Giving to ankle joints angles from 0 to -5 degrees, in 0.5 decrement, the measured global ZMP (according to ZMP double-support equation (2.20) and removing sensors offset) has a linear relation with the set angle. Figure 5.2 shows the obtained average relation expressed as:

$$ZMP_{FT} = q \cdot \theta_A \quad (5.1)$$

where  $ZMP_{FT}$  is the measured ZMP by the Force-Torque sensors,  $\theta_A$  is the angle of the ankle joints and  $q = -0.0166 \frac{m}{deg}$ .

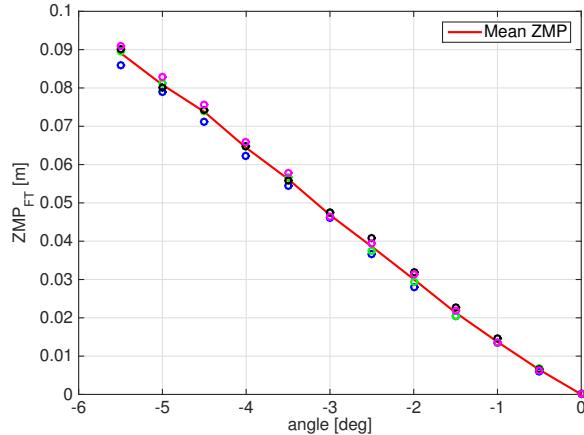


Figure 5.2: Experimental  $\theta_A - ZMP_{FT}$  relation

## 5.2 Position control

The stabilizer is designed to control the ZMP position through commanding different angular positions to robot joint ankles. Then the model parameters obtained in (4.26) and (4.27) are used.

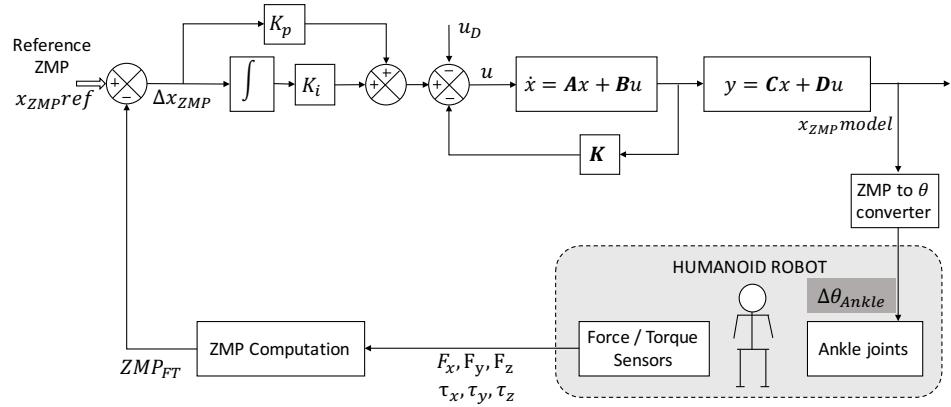
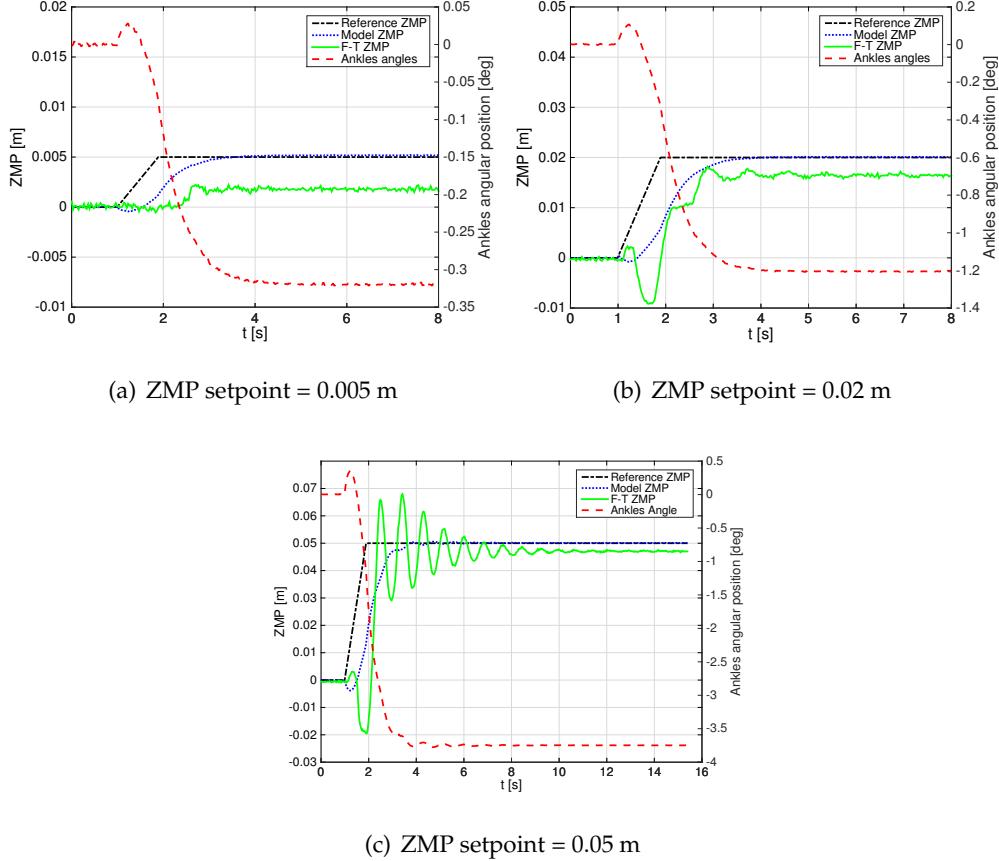


Figure 5.3: Position control diagram

Figure 5.3 depicts the position control diagram followed in this work. Here, the predictive controller estimates a position of the ZMP point, called as  $ZMP_{model}$  using the error from the F-T sensor information ( $ZMP_{FT}$ ) to the setpoint ZMP ( $ZMP_{ref}$ ). This estimated ZMP position is the control signal sent to the robot ankle joints, but position has to be converted into angular units using the converter factor obtained in the previous section. But experimentally, taking the relation above, the  $ZMP_{FT}$  is different from the setpoint value as one can see in Figure 5.4. All figures (a), (b) and (c) have remarkable errors in the ZMP positioning (0.021 m, 0.004 m and 0.003 m respectively). Although the error decreases with the increase of the reference, it has to be analyzed.

The observed error can be caused by various factors. Firstly, the model used for the control system is a linearisation of the Inverted Pendulum Model, and real systems are not completely ideal. The assumption of taking  $\sin \theta = \theta$  for



**Figure 5.4:** Step responses with measured ZMP error in joint position control

small angles, or the lack of accuracy in the location of the CoM, can introduce mathematical errors. Also, there can be measurement deviations in the F-T sensors due to calibration errors, or in analogue to digital conversions. Other systematic errors as the flexibility of the structure (due to the high height of the robot), loosenesses between mechanical parts (as transmissions or unions of pieces), and small irregularities in the ground are not considered, and has no positive effects when talking about a model-based controller. The control signal must be modified in order to take into account the errors previously mentioned. This issue will be discussed in the following sections.

On the other hand, it is obvious from Figure 5.4 that when the desired ZMP increases, overshoot in the transient period increases too. Overshooting may be caused by the structural flexibility and compliance of the robot. We are considering the robot as an inverted pendulum and the real system can differ. First, the robot is not completely rigid, its structure is a little flexible.

Moreover, we are considering a planar surface as the pendulum's base (the sole), not a pivot point. For those reasons, when the ankles angle position changes a bit abrupt, as the velocity is constant, the robot foot base itself balances while the controller tries to reduce the ZMP error.

Then, the control mode is decided to be changed to velocity control. This way, velocity control will provide a better performance of the system response.

### 5.3 Velocity control

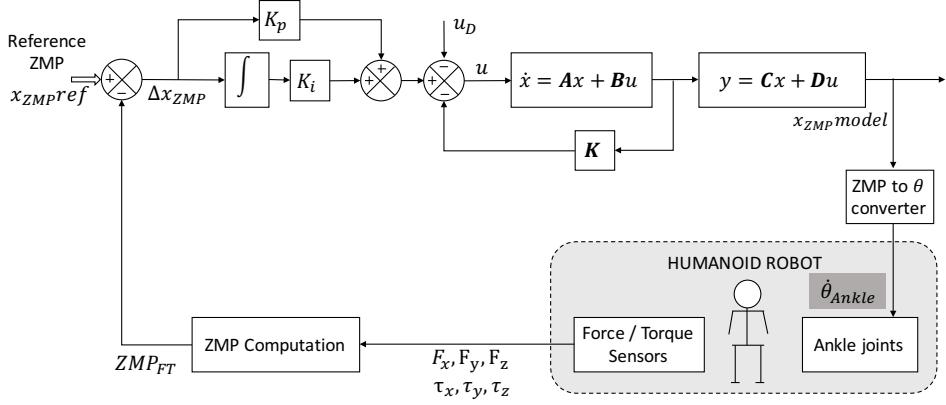
When talking about velocity control, we are talking about the joint control mode. Ankle joints are controlled with angular velocity commands but the ZMP controller and then, the estimated ZMP remains as a position value.

The control diagram presented in the previous section needs to be lightly modified as in Figure 5.5. By its definition, velocity is the rate of change of position with respect to time. Then, angular velocity is obtained as :

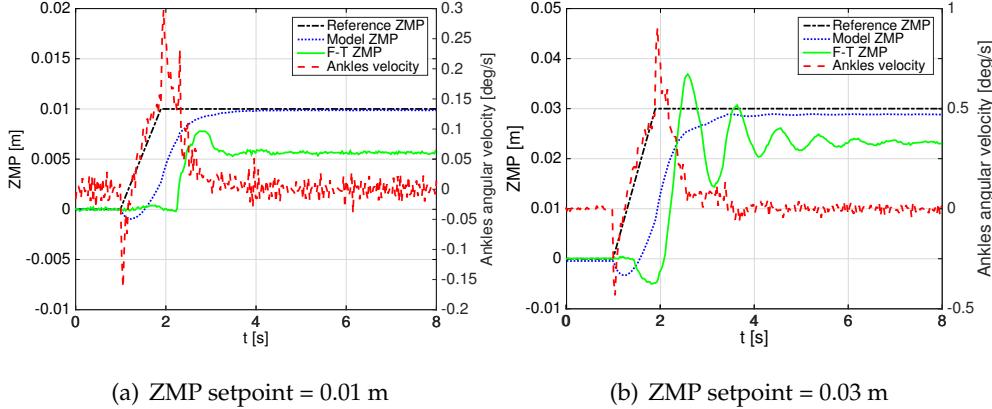
$$\dot{\theta}_k = \frac{\theta_k - \theta_{k-1}}{T} \quad (5.2)$$

where  $\theta_k$  is the actual angular position of the robot ankle joints,  $\theta_{k-1}$  is the previous angular position, and  $T$  is the control loop time (0.03 s).

Figure 5.6 shows the experimental results of ankles velocity control for different setpoint values of ZMP. One can notice that error in system response remains. Also, overshooting in the transient for higher responses is present. As the problem persists, next section will deal with this issue.



**Figure 5.5:** Velocity control diagram

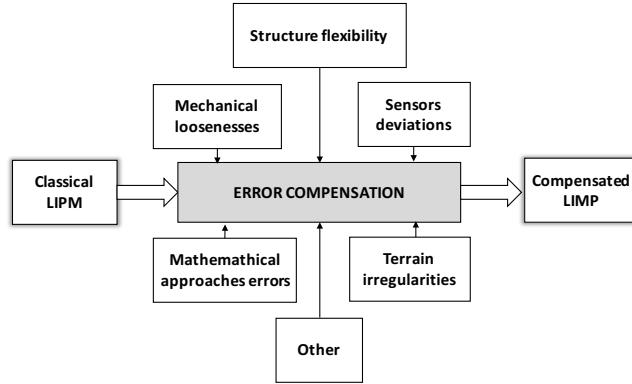


**Figure 5.6:** Step responses with measured ZMP error in joint velocity control

#### 5.4 ZMP error compensation

Now the aim of this work is to improve the classical Inverted Pendulum Model by means of adding system and external error corrections. As mentioned before, model simplifications when controlling real systems have errors. Figure 5.7 summarizes which has been explained before. The purpose of this work is to experimentally obtain errors in the robot performance, try to model them

and make a better approach. Then, the corrected inverted pendulum model will provide a better and robust stability control.



**Figure 5.7:** Error compensation diagram

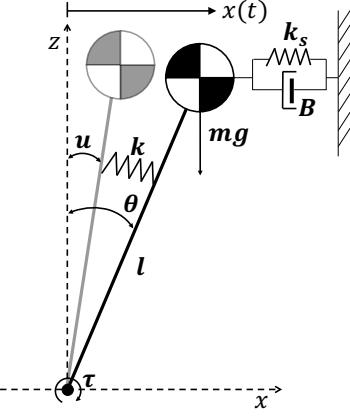
#### 5.4.1 Error compensation model

The Inverted Pendulum Model can be modified by adding mechanical elements as a spring and a damper. In Figure 5.8 one can see the complete system model. The equation of motion of the model is given by:

$$\tau(t) = -ml \frac{d^2}{dt^2}x(t) - Bl \frac{d}{dt}x(t) + mgx(t) - k_s l x(t) \quad (5.3)$$

where  $\tau$  denotes the torque,  $x$  the ZMP (estimated by the model),  $m$  the system mass,  $l$  the pendulum longitude,  $k_s$  the spring and  $B$  the damper constants.  $k$  constant and  $\theta$  and  $u$  angles are only used in the controller design as presented in section 4.2.

On the other hand, the ZMP equation in the  $x$  direction obtained in (2.14) is  $\tau = -x_{FT}mg$  where  $x_{FT}$  is the ZMP obtained from the Force-Torque sensors.



**Figure 5.8:** Proposed Spring-Damper-Mass Inverted Pendulum Model

As the torque  $\tau$  is the input of both equations, the relation between them is obtained by:

$$-ml\ddot{x}(t) - Bl\dot{x}(t) + (mg - k_s l)x(t) = -x_{FT}(t)mg \quad (5.4)$$

If we consider  $x_{FT}$  as a unit step function, the differential equation can be solved for  $x(t)$ . Taking the Laplace transform of the equation above:

$$-mls^2 X(s) - BlsX(s) + (mg - k_s l)X(s) = -\frac{1}{s}mg \quad (5.5)$$

Then, clearing  $X(s)$ :

$$X(s) = \frac{\frac{g}{L}}{s^2 + \frac{B}{m}s + k_s l - mg} \cdot \frac{1}{s} \quad (5.6)$$

Applying the inverse Laplace transform, the solution of the differential equation is obtained as follows:

$$x(t) = P + Qe^{-s_1 t} + Re^{-s_2 t} \quad (5.7)$$

where

$$P = \frac{4m^2 g}{LB^2 - L^2 B^2 - 4m^2 lg + 4l^2 mk_s}$$

$$Q = \frac{2mg}{L(B - \sqrt{\alpha})(\frac{B-\sqrt{\alpha}}{2m} - \frac{B+\sqrt{\alpha}}{2m})}$$

$$R = \frac{2mg}{L(\frac{B-\sqrt{\alpha}}{2m} - \frac{B+\sqrt{\alpha}}{2m})(B + \sqrt{\alpha})}$$

$$\alpha = \frac{LB^2 + 4m^2g - 4lmk_s}{l}$$

$$s_1 = \frac{-B + \sqrt{LB^2 + 4m^2g4mlk_s}}{2m}$$

$$s_2 = \frac{-B - \sqrt{LB^2 + 4m^2g4mlk_s}}{2m}$$

$B$  and  $k_s$  parameters are not known, then  $P, Q, R, s_1$  and  $s_2$  are not known too.

In the steady state, when time goes to infinity, the exponentials of equation (5.7) are zero. This way, the steady state value for a desired ZMP is:

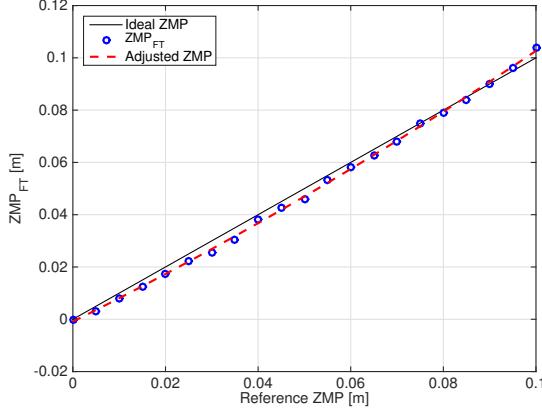
$$x(\infty)|_{x_{REF}} = P \cdot x_{FT}(\infty) \quad (5.8)$$

As  $B$  and  $k_s$  are unknown, an experimental procedure is developed in the next section with the purpose of obtain these parameters values.

#### 5.4.2 Experimental error acquisition

In order to obtain the above unknown parameters, the relation between the a setpoint ZMP value and the measured ZMP by the sensors is experimentally obtained for a wide range of values.

Then, giving ZMP references from 0 to 0.1 m, the ankles angle is obtained from equation (5.1) and the measured ZMP is compared. Figure 5.9 shows the results of the experiment. This relation is obtained in the steady state of both signals. As one can see, as the desired ZMP increases, there always exist an error.



**Figure 5.9:** Reference ZMP and measured ZMP comparative

The measured ZMP can be generalized to a second order polynomial function (using MATLAB Polynomial curve fitting) given by:

$$x_{FT} = ax^2 + bx + c \quad (5.9)$$

and the error can be modelled as:

$$\text{error}_{ZMP} = x - x_{FT} = -ax^2 + (1-b)x - c \quad (5.10)$$

where  $x$  is the Reference ZMP,  $x_{FT}$  is the read ZMP from the sensors,  $a = 1.5682$ ,  $b = 0.8786$  and  $c = -0.0009$ .

### 5.4.3 ZMP error matching

As equation (5.10) is a nonlinear relation, the obtained error is different for each  $x$  operating point. Then, it is proposed a solution based on the *Gain Scheduled Matching*. In control theory, a gain-scheduled controller (*MathWorks website*, 2016) is a controller whose gains are automatically adjusted as a function of time, operating condition, or plant parameters. Gain scheduling is a common strategy for controlling systems whose dynamics change with such variables. Typically,

gain-scheduled controllers are fixed single-loop or multiloop control structures that use lookup tables to specify gain values as a function of the scheduling variables. For tuning purposes, it is convenient to replace lookup tables with parametric gain surfaces. A parametric gain surface is a basis-function expansion whose coefficients are tunable. For applications where gains vary smoothly with the scheduling variables, this approach let us tune a few coefficients rather than many individual lookup-table entries, drastically reducing the number of parameters. This approach also provides explicit formulas for the gains, and ensures smooth transitions between operating points.

In our case, as gain scheduling maps gains in every working point, the *ZMP error matching* reduces ZMP deviations induced by unknown errors. In this case, the variable varying parameter is the error of the ZMP as obtained in (5.10). Therefore, for each  $x$  value, a different error will be considered in the control loop. For each operating point there will be a different value of the  $P$  parameter obtained in (5.7). Later,  $B$  and  $k$  parameters will be obtained according to control specifications as settling time, overshooting and other.

## 5.5 Compensated position and velocity control

Using the control strategy previously defined, the *ZMP error matching* for each operating point, the results of position and velocity control are obtained. Figures 5.10 and 5.11 show different step responses of position control and velocity control respectively. It can be seen that the error decreases from previous experimental trials. Figure 5.12 and Table 5.1 show the improvement in the ZMP measures for a wide range of values.

However, in the transient zone, the ZMP still oscillates and reaches high ZMP values, both in position and velocity control (see 5.10 (c) and (d) and 5.11 (a) and (b)). This high ZMP values can lead the robot to an unbalanced situation if the ZMP overpasses the recovery limit. Then control constraints have to be redefined in order to obtain smoother responses.

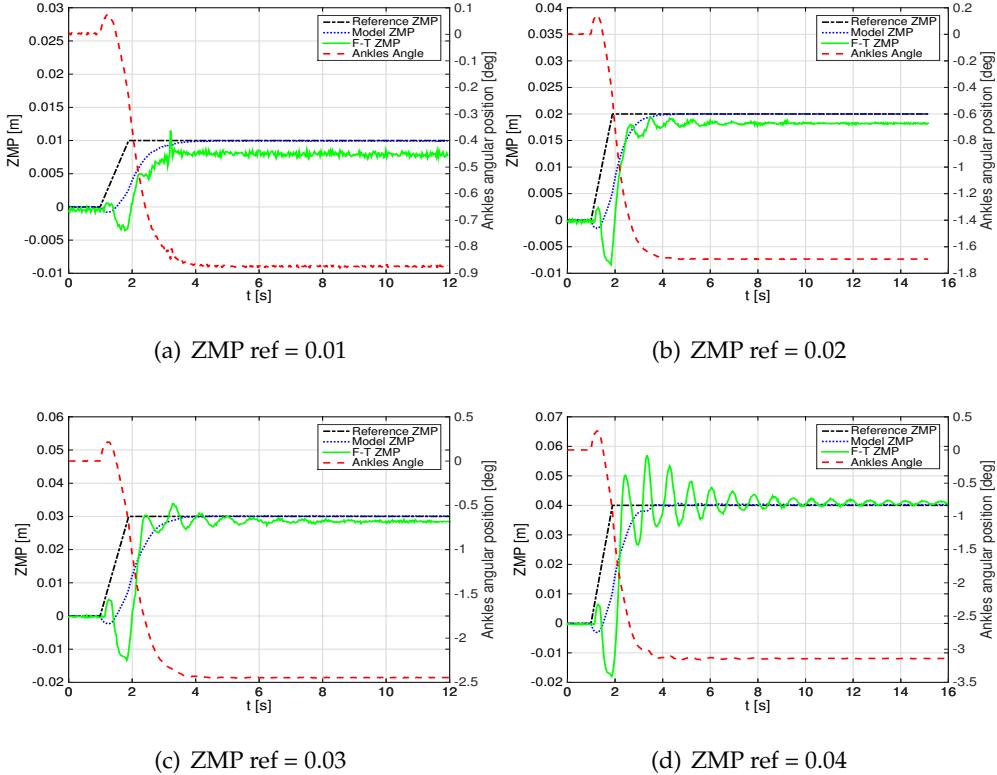


Figure 5.10: Position control step responses

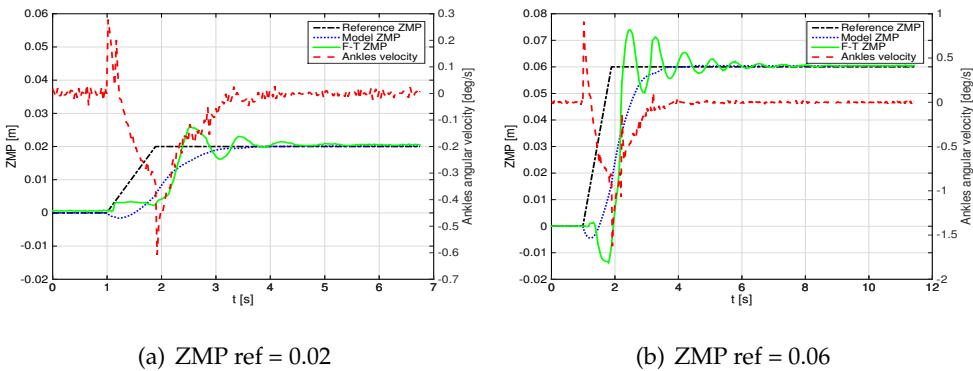
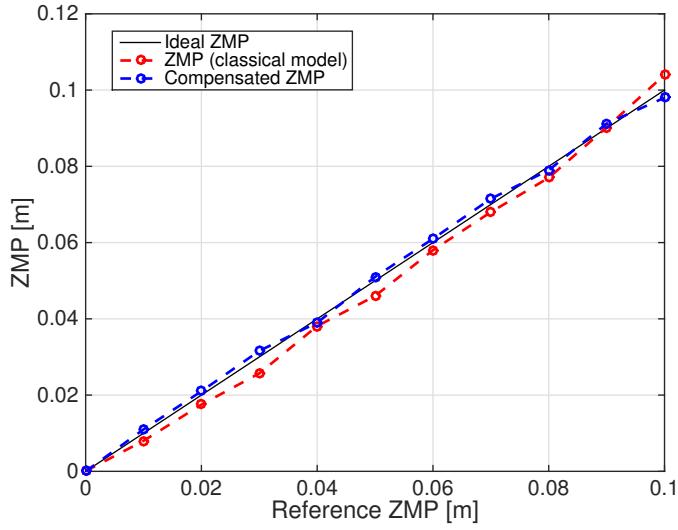


Figure 5.11: Velocity control step responses

**Figure 5.12:** ZMP comparison using Classical and Proposed LIMP

$ZMP_{REF}$ [m]	$ZMP_{FT}$ [m]			
	Classical model	% error	Proposed model	% error
0	$5 \cdot 10^{-5}$	-	$2 \cdot 10^{-7}$	-
0.01	0.0080	20	0.0110	11
0.02	0.0175	12.5	0.0210	5
0.03	0.0257	14.34	0.0316	5.33
0.04	0.0381	4.75	0.0390	2.5
0.05	0.0460	8	0.0510	2
0.06	0.0580	3.34	0.0610	1.67
0.07	0.0680	2.85	0.0715	2.14
0.08	0.0770	3.75	0.0790	1.25
0.09	0.0902	0.22	0.0912	1.33
0.10	0.1040	4	0.0980	2

**Table 5.1:** ZMP comparison using Classical and Proposed LIMP values

On one side, if the control architecture is re-designed using the compensated inverted pendulum model, the transient will be improved. The mechanical parts added to the classical model, will absorb oscillations of the system. On the other side, we have to deal with high ZMP values which can not be compensated using the ankle strategy. In the next section, the ZMP limit for ankle strategy is obtained. We have to ensure that the ZMP position is always below this limit even in the transient or the steady state.

## 5.6 ZMP areas

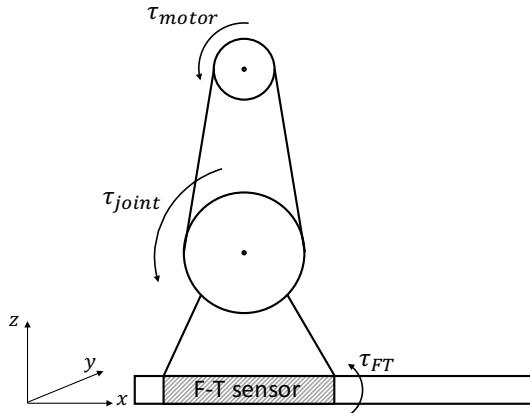
As mentioned before, the robot can recover its balance or not depending on the ZMP position and which parts of its body are compensating the fall down. The ankle strategy should be enough to recover balance from a low disturbance. In order to decide this strategy limit, some experimental trials have been done.

Firstly, the ankle strategy ZMP limit was obtained starting from an upright position -blocking arms, neck and trunk joints-, and giving increasing ZMP reference positions. After several tests, the average ZMP limit forwards is  $0.075m$  and backwards  $-0.02m$ . When ZMP is greater than  $0.075 m$ , the robot can support itself, but the problem comes when recovering. Then  $0.075$  is the maximum ZMP the robot is able to compensate. The great difference of ZMP limit between forwards and backwards is mainly due to the shape of the supporting area (the sole). The mechanical design of the robot feet and legs, make that forwards there is a greater surface in contact with the ground because the center of the ankle joint is displaced rearwards of the center of the sole (see Appendix C).

Secondly, the ankle strategy limit can also be obtained from mechanical features of the ankle motors. Ankle motors are MAXON EC 45 flat, 70 watt brushless motors, 36V. Nominal torque, maximum continuous torque, is  $108 \text{ mNm}$  (see Appendix D). Then, it is not recommended to apply the motor higher torques.

Since we can not access directly to the motor torque, the Force-Torque sensors can be used to achieve that. Figure 5.13 shows the mechanical distribution

of the ankle joint, the motor and the F-T sensor. It is known that the input gearwheel (motor) has 44 teeth and the output gearwheel (joint) has 26 teeth. Then the reduction ratio  $n_1 = z_{input}/z_{output} = 1.69$ . It is also known the gearhead reduction ratio  $n_2 = 160$ .



**Figure 5.13:** Mechanical distribution of the robot's ankle

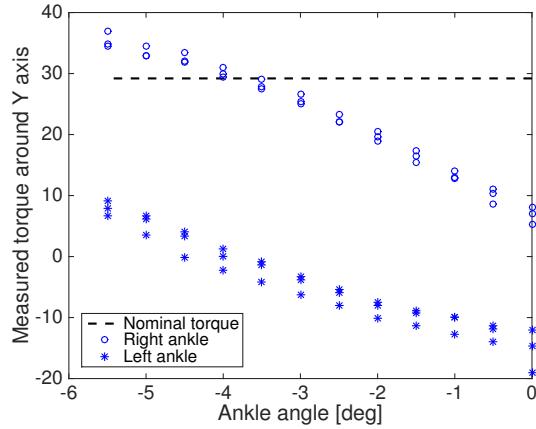
Then, the global reduction ratio is obtained from:

$$\tau_{FT} = \tau_{joint} = 160 \cdot \tau_{gearhead} = 160 \cdot 1.69 \cdot \tau_{motor} = 270.4 \cdot \tau_{motor} \quad (5.11)$$

As nominal motor torque is  $0.108\text{Nm}$ , corresponding measured torque by the sensor is  $\tau_{FT} = 270.4 \cdot 0.108 = 29.20\text{Nm}$

Figure 5.14 shows the measured torques in each foot given by the F-T sensors. It is observed a great difference between right ankle torques and left ankle ones. This difference makes the system to not be accurate and the torque restriction is applied when right ankle exceeds  $29.2\text{Nm}$  while the left ankle motor has not reached its nominal value. Then, a calibration method should be proposed in order to obtain equal torques for the same angle. This way, all the robot weight will be distributed equitably.

Despite this, the ZMP limit is selected using the most restrictive torque (i.e.



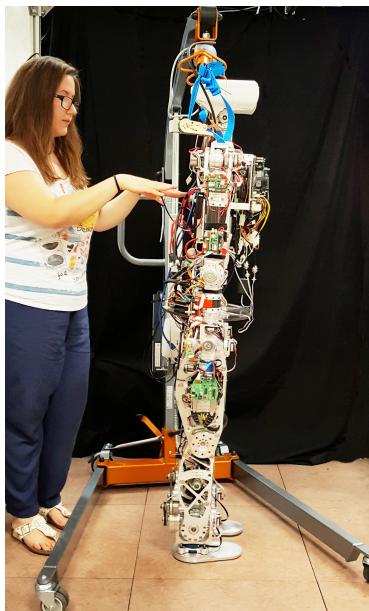
**Figure 5.14:** Measured torques to ankles angles relation

the right ankle torque). Later, after the calibration previously mentioned, maybe this limit changes. The right ankle maximum torque is obtained nearly  $-3.9^\circ$ . This corresponds to  $0.065m$  (using the ZMP-angle relation of equation (5.1)). This ZMP is more restrictive than the one before, then we select this one as the forwards ZMP limit. Summarizing:

$$-0.02m \leq ZMP \leq 0.065m \quad (5.12)$$

## 5.7 Push-Recovery experiments

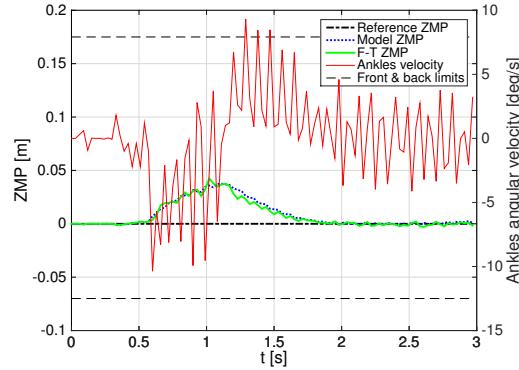
In push recovery experiments, velocity control of ankle joints explained in the previous section is used. But this time, the setpoint value of the ZMP position is always zero. The purpose of this experiments is to always maintain the robot in an upright position when disturbances are applied. The control strategy used to recover the robot balance is the ankle strategy in the double support phase. Figure 5.15 depicts the Push-Recovery experiment when the robot has just been pushed forwards.



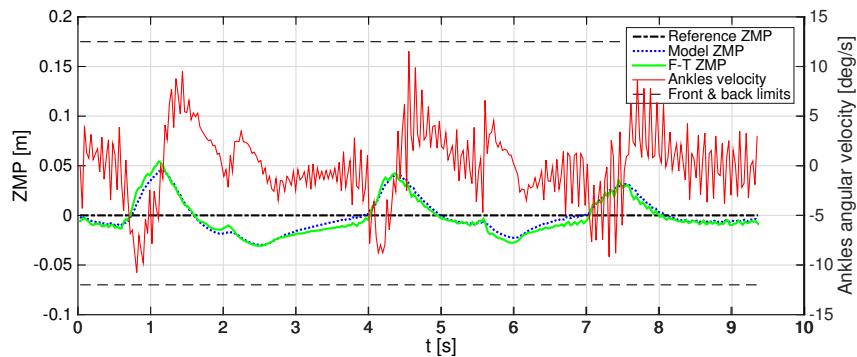
**Figure 5.15:** Push-Recovery experiment performance

Figures 5.16 and 5.17 shows two different experiments in the sagittal plane. Dashed-dot front and back lines delimit the robot support area forwards and backwards. One can see that when a disturbance appears, the control acts commanding a velocity value to ankle joints to recover balance. In both experiments, disturbances are low enough to allow the robot recovery only using its ankles. In 5.16 a pushing movement leads the robot forwards and immediately the re-

covery. In 5.17 a mix of pushing movements are applied and every moment the control system reacts to that disturbances.



**Figure 5.16:** Push-recovery experiment forwards



**Figure 5.17:** Push-recovery experiment forwards and backwards

# Conclusions and future work

## 6.1 Conclusions

This thesis has been focused on the study of balance control for humanoid robots. Nowadays, the development of humanoid robots is a wise bet. Not in a long time, current research will get its benefits. Humanoids development have a great impact on different areas of engineering such as mechanical design, electronics, software and control fields. This work contributes to humanoid research trying to achieve one main premise: humanoid robot TEO has to maintain its balance when low external disturbances are applied. From the overall work carried out trying to accomplish this premise, some conclusions can be extracted:

- The most important criterion for determining the stability of the humanoid robot is the ZMP concept. In order to maintain balance, the ZMP must remain within the robot's support area. Then feedback is essential to close the control loop. When disturbances appear, a proper ZMP correction ensures the robot balance. The Force-Torque sensors used for the ZMP computation have been presented and the design of the data processing system has been proposed.

- Balance has to be ensured at all times. One of the possible ways to achieve that is using a model-based control. The Inverted Pendulum Model uses the dynamics of an inverted pendulum in order to model the humanoid's body. This approach makes the architecture very versatile and adaptable for future modifications and improvements in the robot structure.
- Modelling the robot as an inverted pendulum, the Stabilizer architecture has been proposed. As the robot balance is controlled by means of the ZMP position, the controller is based on a Linear Quadratic Regulator which provides an estimated ZMP position. This estimated ZMP is the Stabilizer control signal applied to the robot joints. The proposed Stabilizer architecture is also versatile and allows to add in the control loop different feedback sensor sources.
- Model approaches deal with a big amount of imprecise information. Pendulum mathematical model is not linear, but both ZMP and model-based control, use a linearised inverted pendulum. Also, errors can be introduced by sensor deviations, mechanical loosenesses and flexible parts in the robot structure, irregularities of the terrain, etc. Then a error compensation model based on the inverted pendulum has been proposed and the results show the improvement.
- From the comparison between position control and velocity control of the ankles at joint level, is velocity control the one that offers better responses. As position control has constant speed the overshoot becomes more remarkable in higher reference values. Furthermore, the mechanical design of the feet can not put up with high overshooting. Robot feet are small and a high overshoots greater than the ZMP limit can lead the robot to an unbalanced situation.
- Finally, this work is focused on the ankle strategy for robot recovery. Therefore some constraints have to be taken into account as the maximum con-

tinuous joint torques. The limit of the ZMP for ankle strategy have been discussed considering this constraint in order to ensure the robot ability to recover its balance.

## 6.2 Future work

The development carried out in this work has been concluded with the ankle strategy balance control of the humanoid TEO. But some interesting research lines have still pending and can be followed up in future work.

It has been only developed the balance ankle strategy. Then the next steps will be to implement the hip and step strategies for recovering from medium and high intensity disturbances. The hip strategy will need to define a new ZMP limit taking again into account joints constraints. The final stability strategy, the step, is more complex. It will be needed to define a gait and ZMP trajectory in order to place the foot in the correct position. When a step is performed, the robot changes to a single support phase. Then, the ZMP computation has to be ensured to take this change into account.

About the ZMP parameter, this work has presented its computation when the robot is in a quasi-static posture. A wide line of research is to obtain the robot arms and torso movements influence in the measured ZMP.

The feedback system used in this thesis are the Force-Torque sensors located at the robot ankles. But the robot also provides an inertial sensor located at its torso. Sensor fusion from this two sources will provide less uncertainty information than these sources used individually. Also, inertial sensor can be used to develop an attitude controller by measuring the velocities and accelerations of the robot body.

Last but not least, errors found in the robot modelling can be minimized removing some of them. Specially it is important to equalize the ankle torques deviation obtained in this work results. Obtaining and correcting sensor deviations, making a better joint calibration of both legs while using the error com-

pensation model proposed will improve the robot performance in stability tasks. Moreover, this work has used a control technique based on gain scheduled controllers to reduce errors in ZMP positioning. Including the compensated inverted pendulum model in the model-based control will ensure high accuracy.

# Bibliography

- Arbulu, M., Pardos, J. M., Cabas, L., Staroverov, P., Kaynov, D., Perez, C., ... Balaguer, C. (2005, December). Rh-0 humanoid full size robot's control strategy based on the Lie logic technique. In *5th IEEE-RAS International Conference on Humanoid Robots, 2005*. (pp. 271–276). doi: 10.1109/ICHR.2005.1573579
- Boston dynamics website.* (2016). url<http://www.bostondynamics.com>.
- Cho, B. K., Park, S. S., & Oh, J. h. (2009, December). Controllers for running in the humanoid robot, HUBO. In *2009 9th IEEE-RAS International Conference on Humanoid Robots* (pp. 385–390). doi: 10.1109/ICHR.2009.5379574
- Gonzalez-Pacheco, V., Ramey, A., Alonso-Martin, F., Castro-Gonzalez, A., & Salichs, M. A. (2011, September). Maggie: A Social Robot as a Gaming Platform. *Int J of Soc Robotics*, 3(4), 371–381. Retrieved 2016-07-16, from <http://link.springer.com/article/10.1007/s12369-011-0109-8> doi: 10.1007/s12369-011-0109-8
- Goswami, A. (1999). Foot rotation indicator (FRI) point: a new gait planning tool to evaluate postural stability of biped robots. In *1999 IEEE International Conference on Robotics and Automation, 1999. Proceedings* (Vol. 1, pp. 47–52 vol.1). doi: 10.1109/ROBOT.1999.769929

Hirai, K., Hirose, M., Haikawa, Y., & Takenaka, T. (1998, May). The development of Honda humanoid robot. In *1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings* (Vol. 2, pp. 1321–1326 vol.2). doi: 10.1109/ROBOT.1998.677288

Kajita, S., Hirukawa, H., Harada, K., & Yokoi, K. (2014). *Introduction to Humanoid Robotics* (Vol. 101). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved 2016-07-16, from <http://link.springer.com/10.1007/978-3-642-54536-8>

Kajita, S., Kanehiro, F., Kaneko, K., Yokoi, K., & Hirukawa, H. (2001). The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. In *2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2001. Proceedings* (Vol. 1, pp. 239–246 vol.1). doi: 10.1109/IROS.2001.973365

Kanehiro, F., Ishiwata, Y., Saito, H., Akachi, K., Miyamori, G., Isozumi, T., ... Hirukawa, H. (2006, October). Distributed Control System of Humanoid Robots based on Real-time Ethernet. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 2471–2477). doi: 10.1109/IROS.2006.281691

Kaneko, K., Kanehiro, F., Kajita, S., Hirukawa, H., Kawasaki, T., Hirata, M., ... Isozumi, T. (2004, April). Humanoid robot HRP-2. In *2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04* (Vol. 2, pp. 1083–1090 Vol.2). doi: 10.1109/ROBOT.2004.1307969

Kaneko, K., Kanehiro, F., Morisawa, M., Tsuji, T., Miura, K., Nakaoka, S., ... Yokoi, K. (2011, September). Hardware improvement of Cybernetic Human HRP-4c for entertainment use. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 4392–4399). doi: 10.1109/IROS.2011.6094415

- Kaynov, D. (2008). *Open motion control architecture for humanoid robots* (Doctoral dissertation). Retrieved 2016-07-16, from <http://e-archivo.uc3m.es/handle/10016/5573>
- Kim, J.-H., & Oh, J.-H. (2004, April). Walking control of the humanoid platform KHR-1 based on torque feedback control. In *2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04* (Vol. 1, pp. 623–628 Vol.1). doi: 10.1109/ROBOT.2004.1307218
- Mathworks website.* (2016). url<http://www.es.mathworks.com>.
- McGeer, T. (1990, April). Passive Dynamic Walking. *The International Journal of Robotics Research*, 9(2), 62–82. Retrieved 2016-07-16, from <http://ijr.sagepub.com/content/9/2/62> doi: 10.1177/027836499000900206
- Metta, G., Fitzpatrick, P., & Natale, L. (2006). YARP: Yet Another Robot Platform. *International Journal of Advanced Robotic Systems*, 1. Retrieved 2016-07-16, from [http://www.intechopen.com/journals/international\\_journal\\_of\\_advanced\\_robotic\\_systems/yarp\\_yet\\_another\\_robot\\_platform](http://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/yarp_yet_another_robot_platform) doi: 10.5772/5761
- Mori, M., MacDorman, K. F., & Kageki, N. (2012, June). The Uncanny Valley [From the Field]. *IEEE Robotics Automation Magazine*, 19(2), 98–100. doi: 10.1109/MRA.2012.2192811
- Nenchev, D. N., & Nishio, A. (2007, October). Experimental validation of ankle and hip strategies for balance recovery with a biped subjected to an impact. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 4035–4040). doi: 10.1109/IROS.2007.4399038
- Ogata, K. (2010). *Modern control engineering*. Prentice Hall.
- Park, I.-W., Kim, J.-Y., Cho, B.-K., & Oh, J.-H. (2008, January). Control hardware integration of a biped humanoid robot with an android

head. *Robotics and Autonomous Systems*, 56(1), 95–103. Retrieved 2016-07-16, from <http://www.sciencedirect.com/science/article/pii/S0921889007001376> doi: 10.1016/j.robot.2007.09.016

Park, I.-W., Kim, J.-Y., Lee, J., & Oh, J.-H. (2005, December). Mechanical design of humanoid robot platform KHR-3 (KAIST Humanoid Robot 3: HUBO). In *5th IEEE-RAS International Conference on Humanoid Robots, 2005*. (pp. 321–326). doi: 10.1109/ICHR.2005.1573587

*The robotcub project website*. (2016). url<http://www.robotcub.org>.

Salichs, M. A., & Malfaz, M. (2012). A New Approach to Modeling Emotions and Their Use on a Decision-Making System for Artificial Agents. *IEEE Transactions on Affective Computing*, 3(1), 56–68. doi: 10.1109/T-AFFC.2011.32

Vukobratovic, M., Borovac, B., & Potkonjak, V. (2007, January). Towards a unified understanding of basic notions and terms in humanoid robotics. *Robotica*, 25(01), 87–101. Retrieved 2016-07-16, from [http://journals.cambridge.org/article\\_S0263574706003031](http://journals.cambridge.org/article_S0263574706003031) doi: 10.1017/S0263574706003031

Vukobratovic, M., Frank, A. A., & Juricic, D. (1970, January). On the Stability of Biped Locomotion. *IEEE Transactions on Biomedical Engineering, BME-17*(1), 25–36. doi: 10.1109/TBME.1970.4502681

Vukobratović, M., & Borovac, B. (2004, March). Zero-moment point — thirty five years of its life. *Int. J. Human. Robot.*, 01(01), 157–173. Retrieved 2016-07-16, from <http://www.worldscientific.com/doi/abs/10.1142/S0219843604000083> doi: 10.1142/S0219843604000083

Winter, D. A., Prince, F., Frank, J. S., Powell, C., & Zabjek, K. F. (1996, July). Unified theory regarding A/P and M/L balance in quiet stance. *Journal of Neurophysiology*, 75(6), 2334–43. Retrieved 2016-07-16, from [https://www.researchgate.net/publication/14409014\\_Unified\\_theory\\_regarding\\_AP\\_and\\_DL\\_balance\\_in\\_quiet\\_stance](https://www.researchgate.net/publication/14409014_Unified_theory_regarding_AP_and_DL_balance_in_quiet_stance)

Appendix A

## JR3 manufacturer information



### Summary of JR3 DSP Data locations

	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07
0x00	ch0time	ch0data			ch1time	ch1data		
0x08	ch2time	ch2data			ch3time	ch3data		
0x38	chEtime	chEdata			chFtime	chFdata		
0x40	'C'	'o'	'p'	'y'	'r'	'i'	'g'	'h'
0x48	't'	' '	'J'	'R'	'3'	'.'	' '	'l'
0x50	'n'	'c'	' '	'1'	'9'	'9'	'4'	0
0x58								
0x60	shunt fx	shunt fy	shunt fz	shunt mx	shunt my	shunt mz		
0x68	def fs fx	def fs fy	def fs fz	def fs mx	def fs my	def fs mz		load env #
0x70	min fs fx	min fs fy	min fs fz	min fs mx	min fs my	min fs mz		xForm #
0x78	max fs fx	max fs fy	max fs fz	max fs mx	max fs my	max fs mz		peak addr
0x80	fs fx	fs fy	fs fz	fs mx	fs my	fs mz	fs v1	fs v2
0x88	ofs fx	ofs fy	ofs fz	ofs mx	ofs my	ofs mz	ofs #	vect axes
0x90	f0 fx	f0 fy	f0 fz	f0 mx	f0 my	f0 mz	f0 v1	f0 v2
0x98	f1 fx	f1 fy	f1 fz	f1 mx	f1 my	f1 mz	f1 v1	f1 v2
0xa0	f2 fx	f2 fy	f2 fz	f2 mx	f2 my	f2 mz	f2 v1	f2 v2
0xa8	f3 fx	f3 fy	f3 fz	f3 mx	f3 my	f3 mz	f3 v1	f3 v2
0xb0	f4 fx	f4 fy	f4 fz	f4 mx	f4 my	f4 mz	f4 v1	f4 v2
0xb8	f5 fx	f5 fy	f5 fz	f5 mx	f5 my	f5 mz	f5 v1	f5 v2
0xc0	f6 fx	f6 fy	f6 fz	f6 mx	f6 my	f6 mz	f6 v1	f6 v2
0xc8	rate fx	rate fy	rate fz	rate mx	rate my	rate mz	rate v1	rate v2
0xd0	min fx	min fy	min fz	min mx	min my	min mz	min v1	min v2
0xd8	max fx	max fy	max fz	max mx	max my	max mz	max v1	max v2
0xe0	near sat	sat	rate addr	rate div	rate count	comm 2	comm 1	comm 0
0xe8	count 1	count 2	count 3	count 4	count 5	count 6	errors	count x
0xf0	warning	error	threshold	crc	rom ver #	ver no	ver day	ver year
0xf8	serial	model	cal day	cal year	units	bits	chans	thickness

0x100-0x1ff - Load envelope table (threshold monitoring), 16 entries

0x200-0x2ff - Transform table (translations and rotations), 16 entries

**Description of table entries, see text for full description and missing entries:**

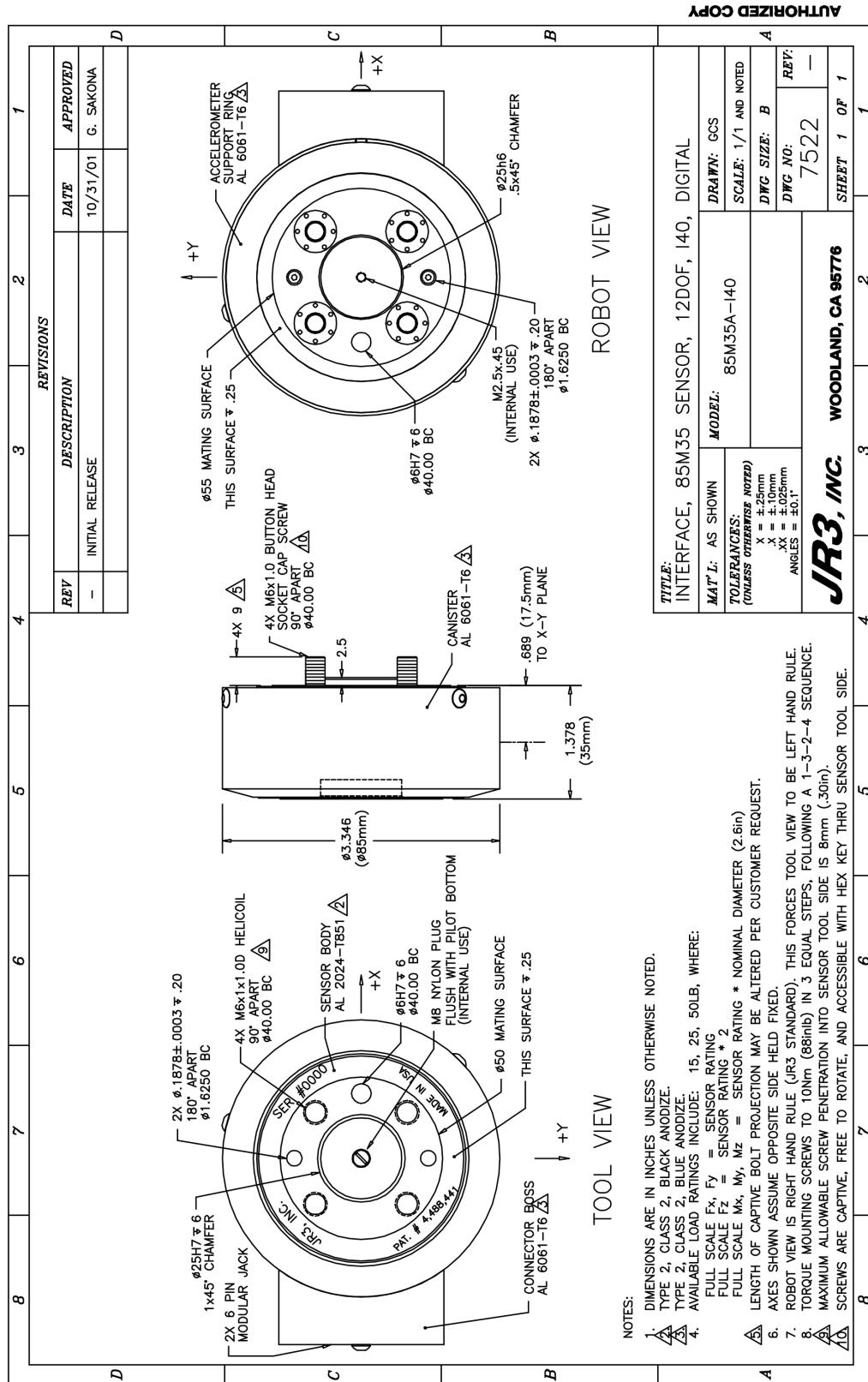
ch0time, ch0data	time last data for channel 0 was received, last data received for raw channel 0
shunt fx,...	shunt reading for fx channel
def fs fx,...	sensor default full scale
min fs fx,...	min full scale, at which the data will not have the lsb zero filled
max fs fx,...	max full scale, at which the data will not have the lsb truncated
fs fx,...	full scale value for fx, when fx = 16384 this is the equivalent engineering units
load env #	number of currently active load envelope
xForm #	number of the transform currently in use
peak addr	addr of the data used in finding the maxima and minima
ofs fx,...	current offset value for fx
ofs #	number of the offset currently in use
vect axes	bit map for the axes which are being used for calculating the vectors
f0 fx,f0 fy,...	decoupled, unfiltered data
f1 fx,...	fx from filter 1
rate fx,...	rate calculation for fx
min fx, ..., max fx,...	minimum peak (valley) value for fx, maximum peak value for fx
near sat, sat	raw value which sets near sat bit in warning word, and sat bit in the error word
rate addr	address of data used for calculating the rate data
rate div	rate divisor, the number of samples between rate calculations
rate count	this counter counts up to rate div, and then the rates are calculated
comm2,...	command word 2, 1 and 0. Area used to send commands to <b>JR3 DSP</b>
count1,...	counter for filter #1, 1 count = 1 filter iteration
errors	a count of data reception errors
warning, error, threshold	warning word, error word, threshold monitoring word (load envelopes)
rom ver no	version no. of data stored in sensor EEPROM
ver no, ver day	software version # that the <b>JR3 DSP</b> is running, <b>JR3 DSP</b> software release date
serial, model	sensor serial number, and sensor model number
cal day	last calibration date of the sensor
units	engineering units of full scale, 0 is lbs, in-lbs and in*1000, 1 is Newtons, ...
bits	number of bits in sensor ADC
chans	bit map of channels the sensor is capable of sending
thickness	the thickness of the sensor



Appendix **B**

JR3 85M35A sensor drawings



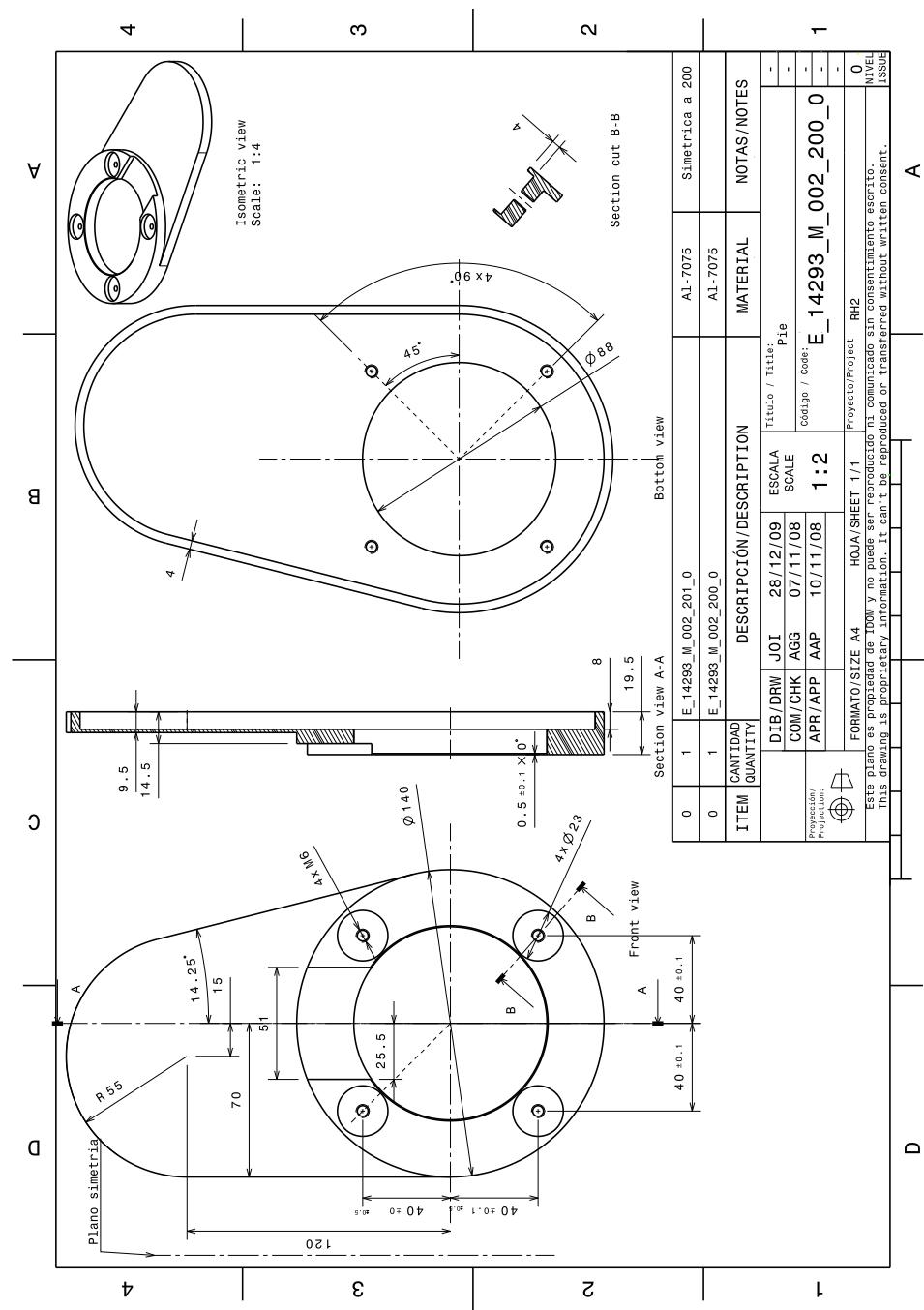




Appendix C

## Foot drawings



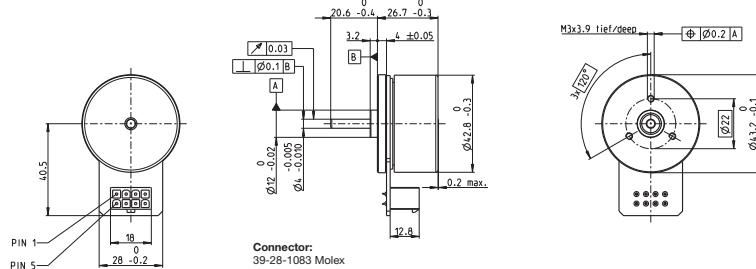




Appendix **D**

## Ankle motors



**EC 45 flat** Ø42.8 mm, brushless, 70 Watt

maxon flat motor

M 1:2

- Stock program
- Standard program
- Special program (on request)

**Part Numbers**

Motor Data (provisional)	with Hall sensors	397172	402685	402686	402687
--------------------------	-------------------	--------	--------	--------	--------

Values at nominal voltage	24	30	36	48	
1 Nominal voltage	V	24	30	36	48
2 No load speed	rpm	6110	6230	6330	3440
3 No load current	mA	234	194	166	48.1
4 Nominal speed	rpm	4860	4990	5080	2540
5 Nominal torque (max. continuous torque)	mNm	128	112	108	134
6 Nominal current (max. continuous current)	A	3.21	2.36	1.93	0.936
7 Stall torque	mNm	1460	1170	1100	915
8 Stall current	A	39.5	25.8	20.7	6.97
9 Max. efficiency	%	85	84	83	84
<b>Characteristics</b>					
10 Terminal resistance phase to phase	Ω	0.608	1.16	1.74	6.89
11 Terminal inductance phase to phase	mH	0.463	0.691	0.966	5.85
12 Torque constant	mNm / A	36.9	45.1	53.3	131
13 Speed constant	rpm / V	259	212	179	72.7
14 Speed / torque gradient	rpm / mNm	4.26	5.44	5.85	3.82
15 Mechanical time constant	ms	8.07	10.3	11.1	7.24
16 Rotor inertia	gcm²	181	181	181	181

**Specifications**

Thermal data			
17 Thermal resistance housing-ambient	3.56 K/W		
18 Thermal resistance winding-housing	4.1 K/W		
19 Thermal time constant winding	29.5 s		
20 Thermal time constant motor	176 s		
21 Ambient temperature	-40 ... +100 °C		
22 Max. winding temperature	+125 °C		

Mechanical data (preloaded ball bearings)			
23 Max. speed	10000 rpm		
24 Axial play at axial load	< 4.0 N	0 mm	
	> 4.0 N	0.14 mm	
25 Radial play		0.14 mm preloaded	
26 Max. radial load (dynamic)	3.8 N		
27 Max. force for press fits (static)	50 N		
(static, shaft supported)	1000 N		
28 Max. radial load, 5 mm from flange	21 N		

**Other specifications**

29 Number of pole pairs	8	
30 Number of phases	3	
31 Weight of motor	141 g	
Values listed in the table are nominal.		

**Connection**

Pin 1	Hall sensor 1*	
Pin 2	Hall sensor 2*	
Pin 3	V <sub>bus</sub> 4.5 ... 18 VDC	
Pin 4	Motor winding 3	
Pin 5	Hall sensor 3	
Pin 6	GND	
Pin 7	Motor winding 1	
Pin 8	Motor winding 2	

\*Internal pull-up (7 ... 13 kΩ) on pin 3

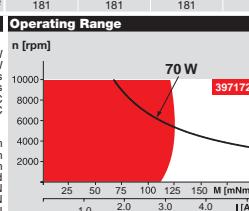
Wiring diagram for Hall sensors see p. 37

**Cable**

Connection cable Universal, L = 500 mm 338980

Connection cable to EPOS, L = 500 mm 354045

April 2016 edition / subject to change

**Comments**

■ Continuous operation  
In observation of above listed thermal resistance (lines 17 and 18) the maximum permissible winding temperature will be reached during continuous operation at 25°C ambient.  
= Thermal limit.

■ Short term operation  
The motor may be briefly overloaded (recurring).

Assigned power rating

**maxon Modular System****Overview on page 20-27**

**Encoder MILE**  
256 - 2048 CPT,  
2 channels  
Page 379

**Recommended Electronics:**

Notes	Page 26
ESCON 36/3 EC	417
ESCON Mod. 50/4 EC-S	417
ESCON Module 50/5	417
ESCON 50/5	418
DEC Module 50/5	420
EPOS2 Module 36/2	424
EPOS2 24/5, 50/5	425
EPOS2 P 24/5	428
EPOS4 Module 50/8	431
EPOS4 Comp. 50/8 CAN	431
MAXPOS 50/5	435

maxon EC motor 301



Appendix **E**

## Guide to the code

The balance control program should be launched following these steps.

- Terminal 1:

ssh 2.2.2.51 (or ssh manipulation)

launch *jr3pci4channelYarp* (Repository: <https://github.com/lpine1/LoliRepo>)

- Terminal 2:

ssh 2.2.2.51 (or ssh manipulation)

launchManipulation

- Terminal 3:

ssh 2.2.2.52 (or ssh locomotion)

launchLocomotion

- Terminal 4:

Launch ankle\_strategy\_sagittal

Code from <https://github.com/lpine1/LoliRepo>