

2-22-2017

Impact Force Reduction Using Variable Stiffness with an Optimal Approach for Jumping Robots

Juan Manuel Calderon Chavez

University of South Florida, juancalderon@mail.usf.edu

Follow this and additional works at: <http://scholarcommons.usf.edu/etd>

 Part of the [Robotics Commons](#)

Scholar Commons Citation

Calderon Chavez, Juan Manuel, "Impact Force Reduction Using Variable Stiffness with an Optimal Approach for Jumping Robots" (2017). *Graduate Theses and Dissertations*.
<http://scholarcommons.usf.edu/etd/6615>

This Dissertation is brought to you for free and open access by the Graduate School at Scholar Commons. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Scholar Commons. For more information, please contact scholarcommons@usf.edu.

Impact Force Reduction Using Variable Stiffness with an Optimal Approach for Jumping Robots

by

Juan M. Calderon

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Electrical Engineering
College of Engineering
University of South Florida

Co-Major Professor: Wilfrido Moreno, Ph.D.
Co-Major Professor: Alfredo Weitzenfeld, Ph.D.
Chung Seop Jeong, Ph.D.
Paris Wiley, Ph.D.
William Quillen, Ph.D.

Date of Approval:
February 8, 2017

Keywords: Falling robot, Humanoid robot, One-legged robot, Optimal control, Fuzzy logic

Copyright © 2017, Juan M. Calderon

DEDICATION

To my Paito. You will be the light of my life forever.

ACKNOWLEDGMENTS

I would like to thank Dr. Wilfrido Moreno and Dr. Alfredo Weitzenfeld for their invaluable guidance and support during the development of this work. I would also like to thank the graduate committee members, Dr. Chung Seop Jeong, Dr. Paris Wiley, and Dr. William Quillen, for taking the time to be a part of my committee and for their valuable feedback. I would like to thank all my classmates and friends, especially, Martin Llofriu, Yueng de la Hoz, Andres Botello and Luis Jaimes, for their support and all the great experiences over the past years.

TABLE OF CONTENTS

LIST OF TABLES	iv
LIST OF FIGURES	v
ABSTRACT	viii
CHAPTER 1 INTRODUCTION	1
1.1 Motivation	2
1.2 Problem Statement	3
1.3 Contributions	3
1.4 Structure of the Dissertation	4
CHAPTER 2 LITERATURE REVIEW	5
2.1 Note to the Reader	5
2.2 Background	5
2.2.1 Humanoid Robot Developments	5
2.2.1.1 Advanced Step in Innovative Mobility (ASIMO)	5
2.2.1.2 Humanoid Robotics Project (HRP)	6
2.2.1.3 Kondo Humanoid Robot (KHR)	6
2.2.1.4 Waseda Bipedal Humanoid (WABIAN)	7
2.2.1.5 NAO	7
2.2.1.6 Dynamic Anthropomorphic Robot with Intelligence-Open Platform (DarWin-OP)	8
2.2.2 Related Work with Jumping Motion in Humanoid Robots	8
2.2.3 Related Work with Falling Humanoid Robots	12
CHAPTER 3 CONCEPTS ABOUT THE JUMPING PROCESS OF HUMANOID ROBOTS	15
3.1 Note to the Reader	15
3.2 Introduction	15
3.3 Humanoid Jumping Process	15
3.3.1 Preparatory Phase	15
3.3.2 Take-off Phase	16
3.3.3 Flight Phase	16
3.3.4 Landing Phase	17
3.4 Robot Falling	17
3.5 Robot Model	20
3.5.1 Kinematic Robot Model	20
3.5.2 Dynamic Robot Model	22

3.6	Motor and Control Model	29
CHAPTER 4	LOW STIFFNESS GENERATION AND JUMPING CONTROL	35
4.1	Note to the Reader	35
4.2	Introduction	35
4.3	Control of the Take-off Phase	35
4.3.1	Vertical Jump Conditions	35
4.3.2	Computed-Torque Control	37
4.3.3	PID Outer Loop	38
4.4	Landing Phase	39
4.4.1	Low Stiffness Generation	40
4.4.1.1	Electrical Motor Model	40
4.4.1.2	Linear Quadratic Regulator	41
4.4.1.3	Optimal PID Design	42
4.4.1.4	Optimal Variable Stiffness Design Using Optimal PID	45
4.4.2	Control of Landing Phase	46
4.4.2.1	Body Inclination Sensor	47
4.4.2.2	Forward Kinematics	49
4.4.2.3	Calculation and Correction of Center of Mass	49
4.4.2.4	ρ Adjustment Calculation	51
4.4.2.5	Optimal PID Recalculation	52
4.4.3	Simulation of the Control Landing System Strategy	52
4.5	Falling Forward Motion	60
4.5.1	Fall Strategy	60
4.5.1.1	Fall Detection	61
4.5.1.2	Arms Position	62
4.5.1.3	Optimal Variable Stiffness Approach	62
CHAPTER 5	OPTIMAL VARIABLE STIFFNESS PERFORMANCE EVALUATION	63
5.1	Note to the Reader	63
5.2	One-legged Jumper Robot Platform	63
5.3	Experiments for Basic Jumping Motion in the Robotic Platform	64
5.3.1	Take-Off Phase	64
5.3.2	Landing Phase	67
5.3.2.1	Landing Phase with Different Values of ρ	68
5.4	Free Fall Experiment	69
5.4.1	Free Fall Experiment with Different ρ Values	69
5.4.2	Free Fall Experiment with Different Height Values	70
5.5	Fall Forward Experiment with Humanoid Robot	71
CHAPTER 6	FUZZY VARIABLE STIFFNESS APPROACH	73
6.1	Note to the Reader	73
6.2	Variable Stiffness with Fuzzy Logic Approach	73
6.2.1	The Fuzzifier	74
6.2.2	Fuzzy Set of Rules	76
6.2.3	Defuzzifier	76

6.3	Experiments for Vertical Jump and Free Fall Motions Using Fuzzy Logic Approach	78
6.3.1	Experiments for Vertical Jump with Constant Impact Velocity and Variable Desired Impact Force	78
6.3.2	Experiment for Free Fall Motion Using Fuzzy Variable Stiffness	79
6.3.2.1	Experiment for Free Fall Motion with Variation of Desired Impact Force Input	79
6.3.2.2	Experiment for Free Fall Motion with Variation of Impact Velocity	80
CHAPTER 7 CONCLUSIONS AND FUTURE WORK		82
7.1	Summary of Results and Findings	82
7.2	Future Work	84
7.2.1	Control Strategies	84
7.2.2	Applications	84
REFERENCES		86
APPENDICES		91
Appendix A	Permission for Reuse	92

LIST OF TABLES

Table 5.1	Parameters of One-legged Robot	63
Table 5.2	Parameters of Figure 5.6	69
Table 5.3	Parameters of Free Fall Experiment with Optimal Approach	71
Table 5.4	Experiment Results	72
Table 6.1	Set of Rules	76
Table 6.2	Parameters of Figure 6.7	79
Table 6.3	Parameters of Free Fall Experiment with Fuzzy Approach	81

LIST OF FIGURES

Figure 2.1	NAO of Aldebaran Robotics.	7
Figure 2.2	Open humanoid platform DARwIn-OP.	8
Figure 2.3	3D-SLIP template model for high-speed humanoid running.	9
Figure 2.4	Phases of jumping motion.	10
Figure 2.5	Pneumatic artificial muscle.	10
Figure 2.6	Initial and optimized vertical jumps for the biarticular robot model.	11
Figure 2.7	NAO robot performing stand up movement using reduced stiffness levels.	12
Figure 2.8	Sequence of three stages in the landing phase of falling animation.	13
Figure 3.1	Vertical jump phases and CoM trajectory	16
Figure 3.2	Martial arts athletes executing rolling motions	18
Figure 3.3	Center of mass projection out of foot support	19
Figure 3.4	Robot model	20
Figure 3.5	Electrical circuit of a typical DC motor.	30
Figure 3.6	PD control model for DC motor	33
Figure 3.7	Step response of motor model with low and high stiffness	33
Figure 4.1	Computed-Torque Control schema with outer PID loop	39
Figure 4.2	PID controller for second order system	43
Figure 4.3	General schema control landing strategy	47
Figure 4.4	Foot angle error in landing phase	48
Figure 4.5	CoM and zero point	50
Figure 4.6	Complete schema control landing	53

Figure 4.7	Joint position with low stiffness	55
Figure 4.8	Joint positions with control landing	56
Figure 4.9	Comparison of joint position with and without control	57
Figure 4.10	Position error for joint with low stiffness with and without control	58
Figure 4.11	Comparison of the CoM position for robot with and without control	59
Figure 4.12	Flow diagram of fall algorithm	61
Figure 4.13	CoM projection	62
Figure 5.1	One-legged jumper robot platform	64
Figure 5.2	Vertical jump performed in the first experiment	65
Figure 5.3	Point zero of foot support	66
Figure 5.4	CoM in take-off phase	66
Figure 5.5	CoM position in a vertical jump using low and full stiffness	67
Figure 5.6	CoM position in vertical jump with different values of ρ .	68
Figure 5.7	CoM position in free fall with different values of ρ .	69
Figure 5.8	CoM position in free fall movement with $\rho = 0.15$.	70
Figure 5.9	CoM position in free fall movement with $\rho = 0.5$.	70
Figure 5.10	Motion sequence of the experiment	71
Figure 5.11	CoM position with different ρ values	72
Figure 6.1	Fuzzy logic system	74
Figure 6.2	Desired-impact force	74
Figure 6.3	Impact-velocity input	75
Figure 6.4	Stiffness output (K_p gain-value)	75
Figure 6.5	Centroid method	77
Figure 6.6	P-Gain value estimation surface	77
Figure 6.7	CoM position in vertical jump with different values of desired impact force input.	78
Figure 6.8	CoM position in a free fall with different values of desired impact force.	80

Figure 6.9	CoM position in free fall movement with desired impact force = 0.3.	80
Figure 6.10	CoM position in free fall movement with desired impact force = 0.7.	81

ABSTRACT

Running, jumping and walking are physical activities that are performed by humans in a simple and efficient way. However, these types of movements are difficult to perform by humanoid robots. Humans perform these activities without difficulty thanks to their ability to absorb the ground impact force. The absorption of the impact force is based on the human ability to vary muscles stiffness.

The principal objective of this dissertation is to study vertical jumps in order to reduce the impact force in the landing phase of the jump motion of humanoid robots. Additionally, the impact force reduction is applied to an arm-oriented movement with the objective of preserving the integrity of falling humanoid robot.

This dissertation focuses on researching vertical jump motions by designing, implementing and testing variable stiffness control strategies based on Computed-Torque Control while tracking desired trajectories calculated using the Zero Moment Point (ZMP) and the Center of Mass (CoM) conditions. Variable stiffness method is used to reduce the impact force during the landing phase. The variable stiffness approach was previously presented by Pratt et al. in [1], where they proposed that full stiffness is not always required. In this dissertation, the variable stiffness capability is implemented without the integration of any springs or dampers. All the actuators in the robot are DC Motors and the lower stiffness is achieved by the design and implementation of PID gain values in the PID controller for each motor. The current research proposes two different approaches to generate variable stiffness. The first approach is based on an optimal control theory where the linear quadratic regulator is used to calculate the gain values of the PID controller. The second approach is based on Fuzzy logic theory and it calculates the proportional gain (K_P) of the PID controller. Both approaches are based on the idea of computing the PID gains to allow for the displacement of the DC motor positions with respect to the target positions during the landing phase. While a DC motor moves from the target position, the robot CoM changes towards a lower position reducing the impact force. The Fuzzy

approach uses an estimation of the impact velocity and a specified desired soft landing level at the moment of impact in order to calculate the P gain of the PID controller. The optimal approach uses the mathematical model of the motor and the factor, which affects the Q matrix of the Linear Quadratic Regulator (LQR), in order to calculate the new PID values.

A One-legged robot is used to perform the jump motion verification in this research. In addition, repeatability experiments were also successfully performed with both the optimal control and the Fuzzy logic methods. The results are evaluated and compared according to the impact force reduction and the robot balance during the landing phase. The impact force calculation is based on the displacement of the CoM during the landing phase. The impact force reduction is accomplished by both methods; however the robot balance shows a considerable improvement with the optimal control approach in comparison to the Fuzzy logic method. In addition, the Optimal Variable Stiffness method was successfully implemented and tested in Falling Robots. The robot integrity is accomplished by applying the Optimal Variable Stiffness control method to reduce the impact force on the arm joints, shoulders and elbows.

CHAPTER 1

INTRODUCTION

In the last three decades, research, development and innovation around humanoid robots have increased exponentially. The improvements of humanoid robots have been evidenced by different organizations around the world. The roadmap for US Robotics: from Internet to Robotics, the Challenge Initiative for Europe Robotics (EuRoC), the Defense Advanced Research Projects Agency (DARPA), and organizations such as RoboCup, have all agreed to focus on the development of humanoid robots as a primary objective for the next 20 years. Some important future applications of humanoid robots are elder care, search and rescue of human victims in disaster zones, human-machine interaction and home assistance, among others.

Humanoid robots require advanced motions and appropriate control strategies to accomplish these applications. RoboCup is a worldwide initiative focused on three main areas: education [2], search & rescue [3] and soccer [4]. The search & rescue area consists of designing robots to search and rescue human victims in disaster zones. The education area encourages children and high school students to work in topics related to science, robotics and technology [5, 6]. Finally, the soccer area focuses in the development of soccer wheeled [7–9] and humanoid robots [10, 11]. In both wheeled and humanoid cases, solving the challenges involves three fields of research: artificial vision [12–14], artificial intelligence [15–19] and locomotion skills [20–22]. The present research work belongs to the third category. Improving jumping movements can help humanoid robots to perform real soccer player actions, such as heading the ball. Also, DARPA [23] encourages the development of humanoid robots that collaborate in the rescue of human casualties in hazardous areas.

The DARPA Humanoid Challenge proposes different tasks, which are focused on the design and the improvement of humanoid robots with new skills and applications related to the real world. To achieve a so-called humanoid-robot interaction in the real world, improving skills such as walking, running and jumping is a must. To improve these locomotion skills, it is necessary to keep the robot balanced while it performs

different movements. With the purpose of keeping balance, Vukobratovic and Borovac introduced in [24] the use of the Center of Mass (CoM) and Zero Moment Point (ZMP) as reference points to perform balance-based control. Additionally, Kajita et al. [25] proposed the inverted pendulum model as a reduced dynamic model for humanoid robots, thus simplifying the design of balance-based control systems for these robots. However, sometimes the balance-based control fails and the robot falls. Some authors describe different approaches for possible solutions to this problem. The consensus is that the main goal is to prevent or reduce damage at the moment of hitting the ground. This research is twofold, first it focuses on the impact force reduction when the humanoid robot is landing while performing a jumping motion and, second, it introduces an approach for minimizing catastrophic robot damage due to unexpected forward falling.

1.1 Motivation

This work proposes that the capability of motor controlled stiffness variability can improve the performance of the prioritized robot skills mentioned above. By varying the motor stiffness, robot locomotion processes could be improved through the impact force absorption when the robots feet touch the ground. To absorb the impact ground force, it is necessary to generate capabilities of variable stiffness in the legs motors of the humanoid robot.

Another important aspect is the preservation of humanoid robots when they lose balance and fall forward. Since 2012, DARPA has been organizing the contest called DARPA Robotics Challenge (DRC), where the development of a robot capable of performing complex tasks in dangerous, degraded, human engineered environments is required. One of the final conclusions of the competition is the necessity of developing strategies to protect the robots when they lose their balance. Usually, these robots can cost up to half a million dollars. Therefore when they fall down, it can cause major financial impact. Impact force reduction can also help emergency movements that aim to protect the robot in an imminent fall. In this dissertation, the approach of using the arms motors stiffness control to protect the robot from catastrophic damage when it hits the ground, will also be presented.

1.2 Problem Statement

According to Pratt [1] full stiffness is not necessary every time. In his work, he proposes the use of low stiffness. He argues in favor of the benefits of low stiffness, including lower reflected inertia, less damage during unexpected contact, stable force control, and shock tolerance. During the last decade, many researchers have focused on the development of variable stiffness using mechanical devices such as springs and dampers. This strategy has allowed for developing low stiffness performance, however the control challenges that arise by the use of springs and/or dampers overcome their benefit. This research proposes the use of low stiffness strategies with the purpose of reducing the impact force during the landing phase of jumping robots and the impact contact for falling robots. The principal objective is the reduction of impact force through the displacement of the motor position with respect to the target. The displacement of the motor positions allows for lowering the robot CoM, which in return decreases the potential and kinetic energies. The variable stiffness is achieved through the variation of the PID gains of the DC motor controllers and without the need to use other components such as springs and/or dampers. The research problem statement is based on the following question: Can variable stiffness approaches using DC motor control reduce the impact force in the jumping process and falling events for biped robots?

1.3 Contributions

The main contributions of this dissertation are presented next. Each one of these contributions will be elaborated in depth in subsequent chapters.

- *Novel approach to reduce the impact force in the landing phase of jumping motions.* The first contribution of this research is proposing a DC motor stiffness control to reduce the impact force in the landing phase of a humanoid robot in the jumping motion.
- *Implementation of an optimal control approach to generation of low stiffness* The second contribution of this dissertation corresponds to a novel implementation of an optimal control strategy to generate low stiffness through the automatic variation of the PID gains in a One-legged robot. The optimal PID gains are designed using a Linear Quadratic Regulator (LQR) method. LQR is capable of keeping balance while reducing the DC Motor stiffness.

- *Low stiffness generation using Fuzzy logic approach.* The third contribution of this dissertation corresponds to an original implementation of a method for stiffness reduction using Fuzzy logic approach in order to optimize computational performance. The method has the aim to estimate the value of proportional gain in a PID controller according to the impact velocity and the the specified desired impact force reduction.
- *Impact force reduction strategy for falling humanoid robots.* The fourth contribution of this dissertation is the proposed control strategy to reduce the impact force when a humanoid robot loses balance and the fall is inevitable. The strategy is based on the positioning of the robot arms in front of head and chest with the aim to protect the main components. Then, the stiffness of elbow and shoulder motors is reduced using a similar approach to the first contribution of this dissertation.

1.4 Structure of the Dissertation

The rest of the dissertation is structured as follows. Chapter 2 contains the related work. Chapter 3 presents basic concepts about the human jumping process, the mathematical model of the robot, and motor model. Chapter 4 presents a control for vertical jump, Computed-Torque Control, and LQR variable stiffness method. Chapter 5 presents the performance evaluation of Optimal Variable Stiffness approach. Chapter 6 presents the Fuzzy Variable Stiffness approach. Finally, chapter 7 presents the conclusions and scope for future research work.

CHAPTER 2

LITERATURE REVIEW

2.1 Note to the Reader

Part of this chapter was published in the IEEE-SoutheastCon [26] and the proceedings of following conference: Innovations in Bio-Inspired Computing and Applications 2015 [27].

2.2 Background

Some of the most important humanoid robots developed in the last three decades are shown in the following section.

2.2.1 Humanoid Robot Developments

In the last three decades, the humanoid robot's development has increased exponentially. Some of the expected applications for humanoid robots are: take care of the elder population, play soccer the same way human beings do, search and rescue of human victims in disaster zones, and home assistance among others. ASIMO, HRP, KAIST, Wabian, NAO, and DarWin are examples of the most important robots developed within the last decades.

2.2.1.1 Advanced Step in Innovative Mobility (ASIMO)

The Honda humanoid robots Asimo is a product of the research done by Honda Robotics Company. It represents one of the most successful examples of a class of bipedal robots. It is built with a design strategy known as precise joint angle control. This robot is fully actuated, which allows for the relative positions of their components to be controlled at all times. It has an advanced walking gait that is smooth and very similar to human walking, turning while walking, running gait, due to its hip joint design. This popular

humanoid robot has inspired many researchers, who used the same type of leg structure, sensors in the feet, and electrical motors with harmonic drives. It has a total of 57 Degrees of Freedom (DoF), Head 3 DoF, Arm 7 DoF x 2, Hands 13 DoF x 2, Hip 2 DoF, and Legs 6 DoF x 2. The robot height is 1.3 m, it weighs 54 kg [28]. The maximum running speed is approximately 9 km/h. The robot is fully equipped with accelerometers, gyro sensors, and force sensor to realize the ZMP feedback walking/running scheme. The robot can predict the next ZMP reference point; which helps to smooth the turning while walking. With fully charged battery, it can operate for up to 60 minutes.

2.2.1.2 Humanoid Robotics Project (HRP)

The Japanese National Institute of Advanced Industrial Science and Technology (AIST) research center has produced many humanoid robots, such as HRP series robots. It is 1.54 m tall and it weighs 58 kg. It has 30 DoF. This robot is the first humanoid built close to a human size. It is used by many researchers to perform research in areas such as walking on uneven surface and operations requiring human interaction. HRP-3 has 42 DoF, is 1.6 m tall, weighs 68 kg, and can operate for 120 minutes with a single charge with normal operation. Their last version HRP-4 is 1.54 m tall, weighs 39 kg, has 34 DoF, and it can operate for up to 45 minutes.

2.2.1.3 Kondo Humanoid Robot (KHR)

The Korea Advanced Institute of Science and Technology (KAIST) has designed many humanoid robots. KHR-1 weighs 48 kg and it is 1.2 m tall. It has 21 degrees of freedom. The KHR-2 model has 41 degrees of freedom, with new sensors such as inertia, tilt, and force torque sensors. The KHR-3 (HUBO) is the latest model, which is designed with 12 DoF in each leg, 8 DoF in each arm and 1 DoF in the torso, for a total of 41 DoF. The HUBO weighs 56 kg and is 1.25 m tall. In addition, it has 10 fingers and 2 cameras. The KHR-3 is built with more human-like features and movements, and the main difference with respect to the KHR-2 is the mechanical stiffness. According to its specifications, it has 50 minutes of autonomous operation.

2.2.1.4 Waseda Bipedal Humanoid (WABIAN)

This humanoid robot is a product of the Waseda University robotics research. Its height is 1.5 m, it weighs about 64 kg. This robot has very smooth and human-like performance in walking. This humanoid has a 2 DoF hip, 6 DoF in each leg, it has 1 DoF in each foot which allow for bending during steady walking. It has a total of 41 DoF. With its flexible pelvis, it has the ability to walk with stretched knees. In addition, it has flexible toes which enables to lift off its toes and support the leg during walking. This humanoid also uses ZMP criteria for stability.

2.2.1.5 NAO

Aldebaran Robotics developed the NAO humanoid robot, see Figure 2.1. The NAO weighs 4.5kg, stands 57 cm high and has 25 degrees of freedom (DOF). NAOs sensors include 2 cameras, 4 microphones, sonar range finders, 2 IR emitters & receivers, 1 inertial board, 9 tactile sensors, and 8 pressure sensors. It has also various communication devices, including voice synthesizer, LED lights, and 2 high-fidelity speakers [29].

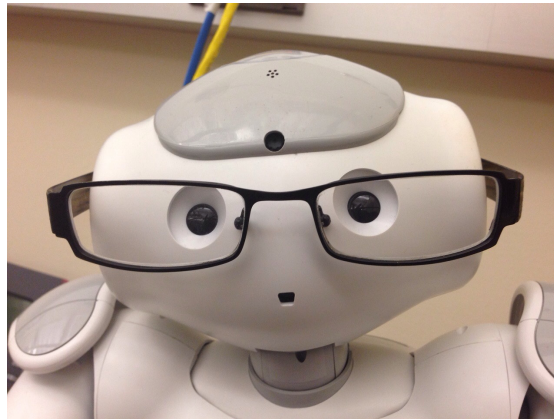


Figure 2.1: NAO of Aldebaran Robotics.

The NAO humanoid has one Intel ATOM 1.6 Ghz CPU, located in the head, that runs a Linux kernel and supports Aldebarans proprietary middleware (NAOqi) and a second CPU which is located in the torso. NAOs 27.6-watt-hour battery provides the NAO with up to 30 minutes of autonomy. There are five DOF in each leg; two in the ankle, two in the hip and one at the knee. An additional degree of freedom exists at the hip; however, it is shared between both legs; that is, both legs are conjointly rotated outward or inward using this joint. The NAO uses brushed DC motors with magnetic rotary encoders for position feedback.

The NAO humanoid also has an open-loop walk capability, where walk patterns can be generated in real-time from a simple ZMP trajectory which is obtained using users specified step parameters such as step frequency, step width, and step length. The ZMP trajectory is then transformed into a Center of Gravity (CoG) based trajectory using an inverted pendulum model.

2.2.1.6 Dynamic Anthropomorphic Robot with Intelligence-Open Platform (DarWin-OP)

DARwIn-OP is a robot designed mainly for research and education purposes at the Robotics & Mechanisms Laboratory (RoMeLa) at Virginia Tech, see Figure 2.2, [30]. It is a completely open source platform; both the hardware and the software can be customized using various software development tools such as C++, Python, LabVIEW, MATLAB, among others.

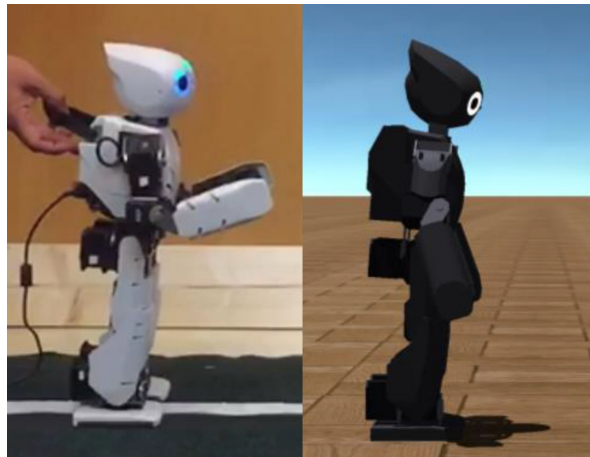


Figure 2.2: Open humanoid platform DARwIn-OP.

2.2.2 Related Work with Jumping Motion in Humanoid Robots

This research extends the original ideas exposed by Kajita et al. in [25], where jumping is used as a first attempt to run, and by Raibert et al. in [31], where a study about balance and control of a legged hopping robot that is able to jump and run was presented. Additional related work includes different aspects of jumping and running models described in [32], [33] and [34]. Reference [32] explains a hopping dynamic control study through simulations of a 26 DoF virtual robot, in which the proposed control system is based on the spring-loaded invert pendulum (SLIP) model. The references [33] and [34], describe a 3D SLIP

model extension, see Figure 2.3, and its model optimization with the aim of enabling the robot to run. These results were verified by simulation.

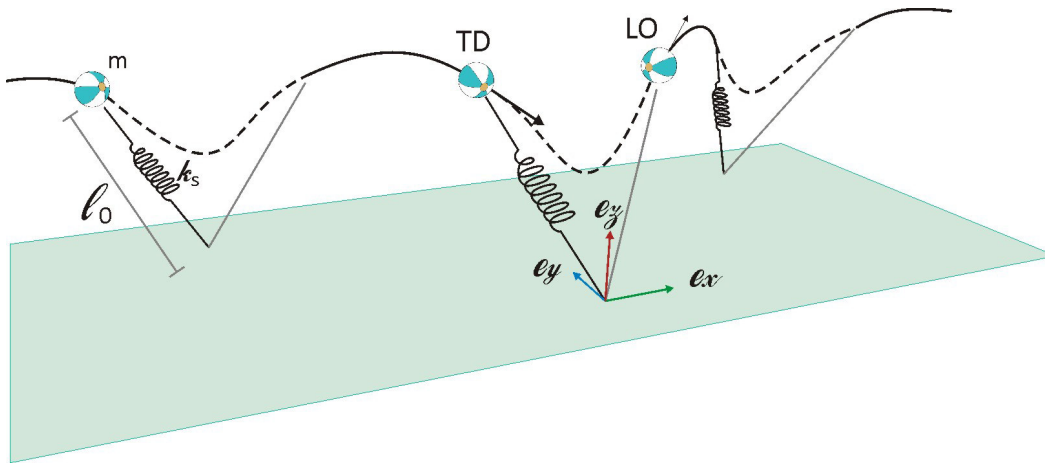


Figure 2.3: 3D-SLIP template model for high-speed humanoid running.

Sakka and Yokoi [35] describe how to use the Ground Reaction Force (GRF), and robot inertia to optimize jump height using a virtual version of an HRP robot. They also propose a motion pattern generation method for vertical jumping in a humanoid robot [36]. In addition, they present a special policy of movement in the landing phase that reduces the impact force. This approach; however, does not use compliant actuators and tests were performed in simulation only.

Nunez et al. [37] present a comparison between jumping movements with and without compliant capability using a virtual HRP robot. They obtained the best performance using compliant features.

Goswami and Vadakkepat [38] use a four-link planar biped robot model with a small compliance capability on the foot plate. They proposed a control model with stable landing, and their compliance was modeled as a spring-damper system. Figure 2.4 shows the jumping motion phases used to simulate the planar biped robot model.

Missura and Behnke [39], describe a robot with compliant motor capabilities where they proposed an algorithm to generate an open-loop walking motion in a bipedal humanoid robot of the RoboCup NimbRo team. While their work focused on the walking movement, they used comparable motors to those used in this work to generate compliant actions.

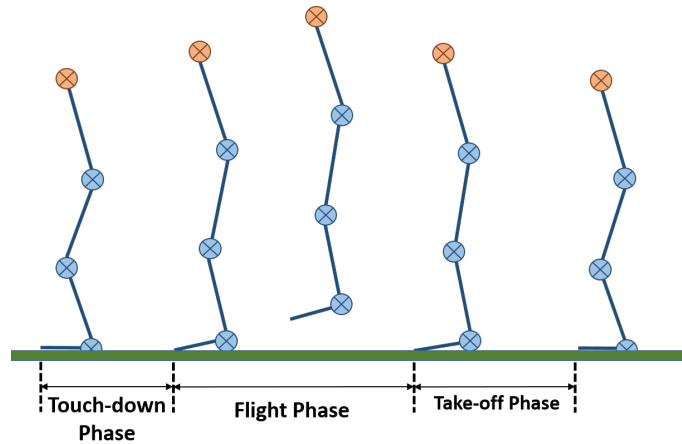


Figure 2.4: Phases of jumping motion.

A Fuzzy logic control approach applied to a real robot called KURMET is presented in [40] and [41]. This robot is a five-link planar biped using Unidirectional Series-Elastic Actuators (USEAs). The actuator consists of a DC brushless motor, and a planetary gearhead in series with a spiral torsion spring. This robotic platform uses a Fuzzy control approach to perform jumping and running movements.

Other types of actuators have been used to perform compliant movements, such as [42] and [43] where authors used pneumatic actuators to create an artificial muscle with similar mechanical properties to those of human muscles, see Figure 2.5.

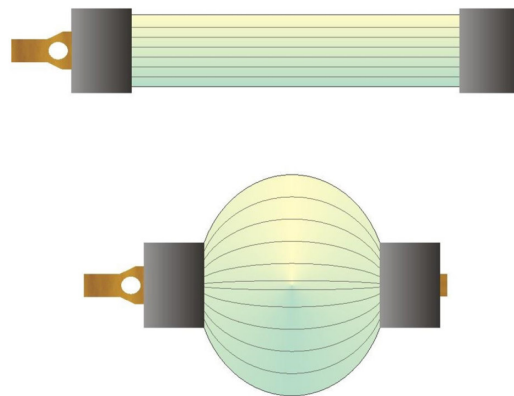


Figure 2.5: Pneumatic artificial muscle.

These papers describe different features of the proposed artificial muscle, including the basic control system that is employed to develop humanoid robots with compliant movements. Similarly, [44] proposes

a trajectory generation method to perform jumping and walking movements in the LUCY robot. This robot was developed using artificial pneumatic muscles as explained previously.

In [45], a torque control method was presented. It performs movements in the LUCY robot using trajectories generated in previous work [44].

In [46] and [47], the authors present studies about responsibility of gastrocnemius muscles and elastic tendons in a jumping process. The researchers evaluated the effect of muscles, tendons, and their stiffness in the height of the vertical jump. This work introduced a robot dynamic model including the stiffness effect, concluding that there is a dependency between the stiffness of the muscle and the ability to jump. This work presents an optimized trajectory for jumping movements based on the relation between stiffness and jumping ability (see Figure 2.6).

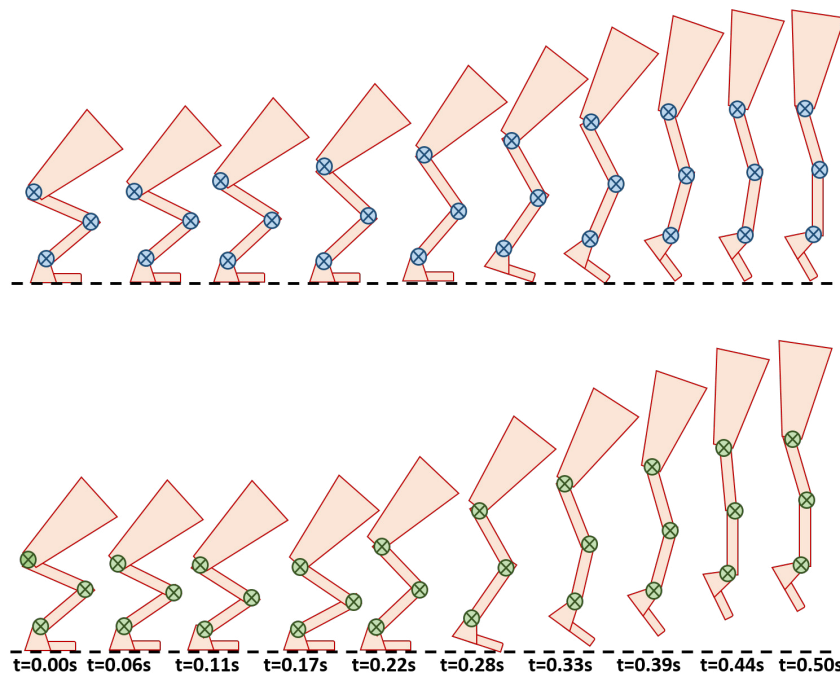


Figure 2.6: Initial and optimized vertical jumps for the biarticular robot model.

The idea that full stiffness is not necessary every time is exposed by Pratt et al. [1]. They argue in favor of the benefits of low stiffness, including lower reflected inertia, less damage during unexpected contact, stable force control, and shock tolerance. Other works such as Elibol et al. [48] and Calderon et al. [49] take advantage of stiffness variation to reduce the energy consumption of a humanoid robot. The first paper

shows the performance variation of a walking process using different joint stiffness values. The second paper proposes a statistic method to calculate the minimum stiffness value required to perform a stand-up movement, see Figure 2.7.



Figure 2.7: NAO robot performing stand up movement using reduced stiffness levels.

2.2.3 Related Work with Falling Humanoid Robots

Falling robots have become one of the most interesting research topics recently. It tries to minimize the damage in the robot joints and the important parts such as chest and head, where the processors, cameras, and batteries are usually located. Some previous work proposes different ways to reduce the impact force in falling robots, such as fall prediction, fall sequences generation, reducing the shock force exerted in the ground impact, mechanical improvements, and manipulation of joint stiffness.

One of the most important steps in the study on the falling robot is to predict and detect when the robot is falling [50]. Karsen and Wirsén [51] predict fall events using principal component analysis.

The most popular technique to reduce the damage on the robot at the moment of impact is to take inspiration from what the human reaction would be in different falling scenarios. Fujiwara et al. [52] and [53] make decisions based on martial arts or more specifically judo techniques, while also trying to reduce the angular momentum.

Ruiz-del-Solar [54] uses a similar approach based on Japanese martial arts skills. However he adds another concept based on the idea of keeping the CoM as low as possible to reduce the impact force on the joints.

Wilken et al. [55] proposed an algorithm for a diving motion in a goalkeeper robot. This algorithm optimizes the falling time and defines the movement trajectory according to the direction and velocity of the ball.

Ha et al. [56] is focused in another interesting approach to the falling motion field. It consists of the use of techniques from animation to reduce impact forces in falling bodies. Figure 2.8 shows an animation where the landing phase of falling motion is divided into three stages: impact, rolling, and getting up.

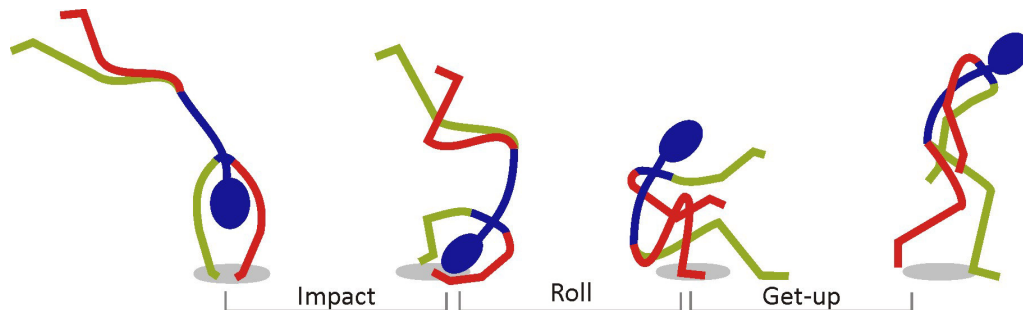


Figure 2.8: Sequence of three stages in the landing phase of falling animation.

Hu and Liu et al. [57] also show different research approach to reduce the damage in humanoid falls. They propose an algorithm focused on the dissipation of the momentum in the initial phase of the fall. They do so by using multiple contact points with the ground, thus splitting the impact force through various contact points. They validated their algorithm using physics simulation software and a BioloidGp humanoid robot.

Wilken et al. [55] study the case of a goalkeeper robot, they proposed to add mechanical improvements to the more exposed parts, susceptible to damage due to falls, such as the hip and the upper limbs. The goal is to enlarge the lifespan of the joints. Thus they propose the use of springs to recover the target position of the joints and the addition of something pads to dampen the shock as well. However, Pratt et al [1] proposed the idea that full stiffness is not always the best way to work. They make a mathematical study about the effect of low stiffness in a robotic joint and how to control it. Additionally, they list several cases in which low stiffness has a good performance, such as stable force control, lowering reflecting inertia, fewer damages

during unexpected contact and shock tolerance. Shock tolerance and unexpected contact are the two main arguments in favor of the use of variable stiffness in the present work.

In this research, the use of low stiffness to protect the robot joints during falling motions or in the landing phase of the jumping process is being proposed. The proposed research introduces a complete framework to perform jumping motions with biped robots. It is based on the generation of low stiffness in DC motor joints via PID controller gains computation. The low stiffness is applied to reduce the impact force in the landing phase of jumping motions and falling strategies for biped robots.

CHAPTER 3

CONCEPTS ABOUT THE JUMPING PROCESS OF HUMANOID ROBOTS

3.1 Note to the Reader

Part of this chapter was published in the IEEE-SoutheastCon [26] and the proceedings of following conference: Innovations in Bio-Inspired Computing and Applications 2015 [27].

3.2 Introduction

This chapter introduces the basic concepts about the jumping process in humans and how it can be applicable to humanoid robots. Additionally, the mathematical model of the humanoid robot and DC motor are depicted.

3.3 Humanoid Jumping Process

Vertical jumping is the action executed by human beings when the Center of Mass (CoM) is raised over the normal stand-up human position. This movement has to be performed solely by the muscle actions without the help of any external device. The main criterion to evaluate vertical jump efficiency is the maximum height reached in the flight phase. The vertical jump is normally divided into four phases: preparatory, take-off, flight and landing, as depicted in Figure 3.1.

3.3.1 Preparatory Phase

It is when the CoM is moved to a lower position, and the potential energy decreases. The hips and knees are flexed, and the ankles are dorsiflexed.

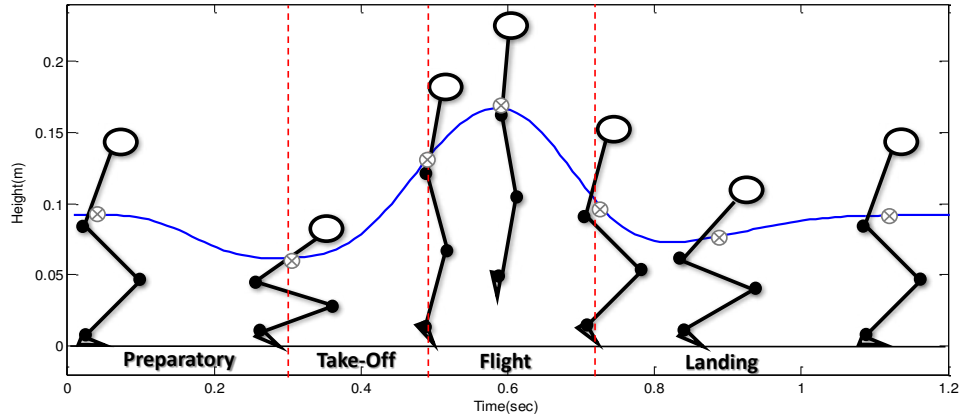


Figure 3.1: Vertical jump phases and CoM trajectory

3.3.2 Take-off Phase

It is when the CoM is moved to a higher position, and the potential energy increases. The hips and knees are extended and the ankle plantar are flexed. During this phase, the feet stay in contact with the ground. This phase finishes when the feet are no longer touching the ground.

3.3.3 Flight Phase

The flight phase is when the body is in the air. It starts when the feet are no longer in contact with the ground, and it finishes when they touch the ground again. The height of the jump depends on the velocity reached by the CoM at the beginning of the phase. During the flight phase, the body loses control of the rotation and trajectory, thus the position of the body is determined by the trajectory, velocity, and acceleration of the previous phase. Acceleration, velocity, position, and maximum height in the flight phase are described by Eqs. (3.1) through (3.4), respectively.

$$\ddot{y}_{com}(t) = -g \quad (3.1)$$

$$\dot{y}_{com}(t) = -gt + V_{to} \quad (3.2)$$

$$y_{com}(t) = -\frac{1}{2}gt^2 + V_{to}t + Y_{to} \quad (3.3)$$

$$Y_{max} = -\frac{V_{to}^2}{2g} \quad (3.4)$$

where g is the gravity force, V_{to} and Y_{to} are the velocity and position at the end of the take-off phase, respectively.

3.3.4 Landing Phase

It is when the feet touch the ground again. In this phase, the lower body tries to absorb and reduce the impact force exerted by the floor. Assuming that the robot velocity before the impact (landing velocity) is known, the impact force can be estimated as:

$$F_{i-avg} = \frac{\frac{1}{2}mV_l^2}{d} \quad (3.5)$$

where m is the mass of the body, V_l is the landing velocity, which can be computed using Eq. (3.2) and d represents the distance traveled by the robot CoM after the impact. Figure 3.1 shows the position of a human CoM during a vertical jump movement. The trajectory shows how the CoM is going down in the landing phase when the legs are trying to damp the impact force exerted by the ground. A similar approach is employed in this work, where the landing impact is naturally reduced through the use of low stiffness in the electrical motors driving the ankle and knee joints.

3.4 Robot Falling

The analysis and simulation of falls are very important topics in bio-mechanical research. This type of research suggests that the arms and legs in a fall do not move randomly and try to execute movements to land safely. Human fall strategies can inspire bio-mimetic strategies for a humanoid robot. The most common fall strategies that humans perform are aimed at minimizing injuries and have two major goals:

- Reducing impact velocity.

- The distribution of impact force in the largest possible area, because fall strategies are a task where time is critical, its success depends on how quickly it is detected that the fall is inevitable and when a reaction action must be performed. This indicates that the reaction time is an important factor to generate safe fall strategies.

Some of the most common strategies to minimize robot damages from falls are:

- Extending the arms to distribute the impact energy.
- Bending the elbows to decrease impact force.
- Landing with the knee to reduce the momentum of failure before the arms make contact with the surface
- Reduce the CoM height to reduce impact velocity and absorb some of the kinetic energy using the lower extremities during descent.
- Perform similar movements to the Martial Arts athletes executing rolling motions as shown in Figure 3.2.

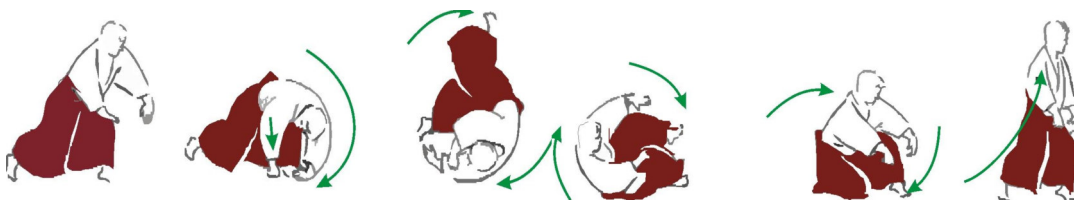


Figure 3.2: Martial arts athletes executing rolling motions

The strategies used by humans to reduce fall damages provide a good approach to the humanoid robot falling. However, not all the techniques used by humans can be applied to humanoid robots. Some human techniques are based on experience and learning. In most cases, humans protect the face and head from the hit. The robots should protect places in the body where the most expensive components such as cameras, sensors, and control systems are located. These devices are not always located in the head of the robot. Another important aspect concerns the differences in the materials human beings and robots are made of. In the case of human beings, the arms and legs are more susceptible to breakage, since the bones are not so

resistant to the blows. In contrast, the robots have links built with hard materials that help them to withstand the blows and the weak points are the unions where the actuators are prone to rupture.

The Simple Linear Inverted Pendulum (SLIP) [25] is the most used model to implement control strategies in humanoid robots. The robot balance is based on maintaining the ZMP and projection of the CoM on the x-axis within the support surface (usually the feet of the robot) as shown in Figure 3.3.

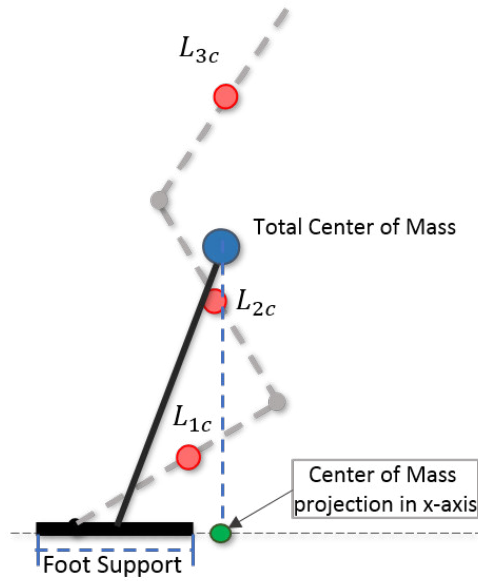


Figure 3.3: Center of mass projection out of foot support

Some important measures to evaluate the performance of the impact force reduction algorithms are Impact Force Average described in Eq. (3.5).

Here, the definition of impact force is kept as in the previous section. Where $V_l = \dot{y}_{com}(t)$ is the landing velocity or impact velocity according with the case. It is expressed by Eq. (3.6), where V_{to} can be zero or some initial velocity value according to the disturbance force causing the falling motion. Finally, d is the displacement of the robot CoM from the touchdown position in the impact moment. The distance is inversely proportional to the impact force.

$$V_l(t) = \dot{y}_{com}(t) = -gt + V_{to} \quad (3.6)$$

3.5 Robot Model

In the human vertical jump, both legs are doing the same movements. In accordance to this, a reduced planar model with three Degrees of Freedom (DoF) is proposed. The proposed model has three joints and four links. The joints are the ankle, the knee, and the hip. The links are the foot, the shank, the thigh and the trunk.

3.5.1 Kinematic Robot Model

The kinematic model is used to calculate the position of every link, and to estimate the velocity, acceleration, and position of the whole robot's CoM, as depicted in Figure 3.4.

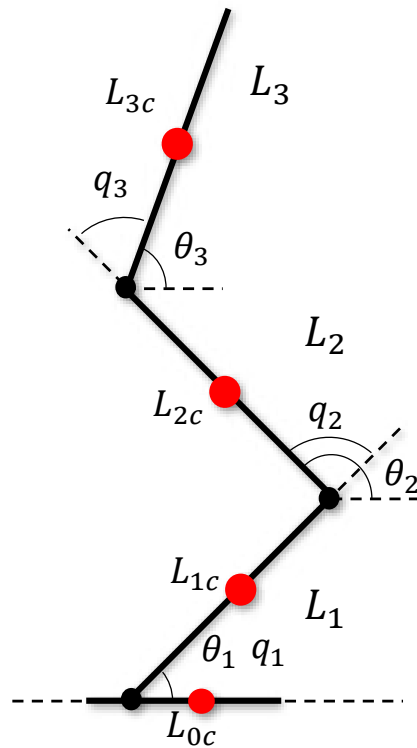


Figure 3.4: Robot model

In Figure 3.4 L_i denotes the length of link i , θ_i is the absolute rotation of joint i , q_i is the relative rotation of joint i with respect to the previous joint i , L_{ic} shows the position of CoM for the corresponding link i , the red circles represent the CoM of every link, and the black circles are robot joints, ankle, knee, and hip.

The Kinematic model of the system is defined by the CoM of each link denoted by x_{ic} which is in the x axis and y_{ic} which is in the y axis as shown in Eqs. (3.7) to (3.12).

$$x_{1c} = L_{1c}\cos(\theta_1) \quad (3.7)$$

$$y_{1c} = L_{1c}\sin(\theta_1) \quad (3.8)$$

$$x_{2c} = L_1\cos(\theta_1) - L_{2c}\cos(\theta_2) \quad (3.9)$$

$$y_{2c} = L_1\sin(\theta_1) + L_{2c}\sin(\theta_2) \quad (3.10)$$

$$x_{3c} = L_1\cos(\theta_1) - L_{2c}\cos(\theta_2) + L_{3c}\cos(\theta_3) \quad (3.11)$$

$$y_{3c} = L_1\sin(\theta_1) + L_{2c}\sin(\theta_2) + L_{3c}\sin(\theta_3) \quad (3.12)$$

The CoM coordinates x_{com} and y_{com} of the whole robot in general terms is written in Eq. (3.13).

$$x_{com} = \frac{\sum_{i=1}^n m_i x_i}{\sum_{i=1}^n m_i}, \quad y_{com} = \frac{\sum_{i=1}^n m_i y_i}{\sum_{i=1}^n m_i} \quad (3.13)$$

where m_i is the mass of each link, x_i and y_i are the absolute position of the link i . The CoM coordinates of the model can be expanded in the x and y directions as shown in Eqs. (3.14) and (3.15).

$$x_{com} = \frac{m_1 x_{1c} + m_2 x_{2c} + m_3 x_{3c}}{m_1 + m_2 + m_3} \quad (3.14)$$

$$y_{com} = \frac{m_1 y_{1c} + m_2 y_{2c} + m_3 y_{3c}}{m_1 + m_2 + m_3} \quad (3.15)$$

The Linear velocity of the CoM for each link can be written as shown in Eqs. (3.16) to (3.18). For link 1, the CoM position was defined by Eqs. (3.7) and (3.8), then the linear velocity for the CoM of the link 1 is:

$$v_{c1} = \begin{bmatrix} L_{1c}\sin\theta_1 \\ L_{1c}\cos(\theta_1) \end{bmatrix} \dot{\theta}_1 \quad (3.16)$$

For link 2, the position of the CoM was shown by Eqs. (3.9) and (3.10), then the linear velocity of the CoM of the link 2 is:

$$v_{c2} = \begin{bmatrix} L_1\sin(\theta_1) \\ L_1\cos(\theta_1) \end{bmatrix} \dot{\theta}_1 + \begin{bmatrix} L_{2c}\sin(\theta_2) \\ L_{2c}\cos(\theta_2) \end{bmatrix} \dot{\theta}_2 \quad (3.17)$$

For link 3, the position of the CoM was depicted by Eqs. (3.11) and (3.12), then the linear velocity of the CoM of the link 3 is:

$$v_{c3} = \begin{bmatrix} L_1\sin(\theta_1) \\ L_1\cos(\theta_1) \end{bmatrix} \dot{\theta}_1 + \begin{bmatrix} L_2\sin(\theta_2) \\ L_2\cos(\theta_2) \end{bmatrix} \dot{\theta}_2 + \begin{bmatrix} L_{3c}\sin(\theta_3) \\ L_{3c}\cos(\theta_3) \end{bmatrix} \dot{\theta}_3 \quad (3.18)$$

3.5.2 Dynamic Robot Model

The dynamic model is given by the Lagrange-Euler formulation. This equation can be written in the general form, as shown in Eq. (3.19).

$$\frac{d}{dx} \left(\frac{\partial L}{\partial \dot{\theta}_i} \right) - \frac{\partial L}{\partial \theta_i} = \tau_i \quad (3.19)$$

where L represents the Lagrangian equation and τ_i represents the torque for each joint, which includes the control torques, forces and friction effects. The Lagrangian is defined in the following Eq. (3.20).

$$L = K_E - P_E \quad (3.20)$$

where K_E represents the kinetic energy and P_E represents the potential energy of each link. Substituting Eq. (3.20) into Eq. (3.19), Eq. (3.21) is obtained:

$$\frac{d}{dx} \left(\frac{\partial K_E - P_E}{\partial \dot{\theta}_i} \right) - \frac{\partial K_E - P_E}{\partial \theta_i} = \tau_i \quad (3.21)$$

The final form of the Euler-Lagrange equation is depicted in Eq. (3.22) :

$$\frac{d}{dx} \left(\frac{\partial K_E}{\partial \dot{\theta}_i} \right) - \frac{\partial K_E}{\partial \theta_i} + \frac{\partial P_E}{\partial \theta_i} = \tau_i \quad (3.22)$$

The kinetic and potential energy can be written in Eqs. (3.23) and (3.28):

$$K_E = \sum_{i=1}^3 (K_{Ei}) = \sum_{i=1}^3 \left(\frac{1}{2} m_i v_{ci}^2 + \frac{1}{2} I_i \dot{\theta}_i^2 \right) = \sum_{i=1}^3 \left(\frac{1}{2} m_i v_{ci}^2 + \frac{1}{2} I_i w_{ci}^2 \right) \quad (3.23)$$

where I_i represents the inertia of each link, v_{ci} is the linear velocity of the CoM for link i , and w_{ci} is the angular velocity of the CoM for link i . The potential energy is defined by Eq. (3.28).

$$P_E = \sum_{i=1}^3 P_{Ei} = \sum_{i=1}^3 m_i \cdot g \cdot y_{ci} \quad (3.24)$$

where g shows the gravitational acceleration. Following equations show the relationship between linear and angular velocities with the Jacobian matrix as shown by Eqs. (3.25) and (3.26).

$$v_{ci} = J_L^{(i)} \dot{\theta} \quad (3.25)$$

$$w_{ci} = J_A^{(i)} \dot{\theta} \quad (3.26)$$

where $J_L^{(i)}$ and $J_A^{(i)}$ are the Jacobian matrices for linear and angular velocities of link i , respectively. The total kinetic energy of the 3 link model is found in Eq. (3.27):

$$\begin{aligned} K_E = & \frac{1}{2} m_1 L_{1c}^2 \dot{\theta}_1^2 + \frac{1}{2} I_1 \dot{\theta}_1^2 + \frac{1}{2} m_2 L_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_3 L_1^2 \dot{\theta}_1^2 \\ & + \frac{1}{2} m_2 L_{2c}^2 \dot{\theta}_2^2 + \frac{1}{2} I_2 \dot{\theta}_2^2 + \frac{1}{2} m_3 L_2^2 \dot{\theta}_2^2 + \frac{1}{2} m_3 L_{3c}^2 \dot{\theta}_3^2 \\ & + \frac{1}{2} I_3 \dot{\theta}_3^2 + m_2 L_1 L_{2c} \cos(\theta_1 + \theta_2) \dot{\theta}_1 \dot{\theta}_2 + m_3 L_1 L_2 \cos(\theta_1 + \theta_2) \dot{\theta}_1 \dot{\theta}_2 \\ & + m_3 L_2 L_{3c} \cos(\theta_3 + \theta_2) \dot{\theta}_2 \dot{\theta}_3 + m_3 L_1 L_{3c} \cos(\theta_1 + \theta_3) \dot{\theta}_1 \dot{\theta}_3 \end{aligned} \quad (3.27)$$

The total potential energy of the 3 links is computed using Eq. (3.28):

$$\begin{aligned}
 P_E = & m_1 g L_{1c} \sin(\theta_1) + m_2 g L_1 \sin(\theta_1) + m_3 g L_1 \sin(\theta_1) \\
 & + m_2 g L_{2c} \sin(\theta_2) + m_3 g L_2 \sin(\theta_2) + m_3 g L_{3c} \sin(\theta_3)
 \end{aligned} \tag{3.28}$$

The torque formulation for each link is included below, in accordance to calculations by following the general Lagrange expression in Eq. (3.19).

The torque for link 1 is:

$$\begin{aligned}
 \tau_1 = & (m_1 L_{1c}^2 + I_1 + m_2 L_1^2 + m_3 L_1^2) \ddot{\theta}_1 + (m_2 L_1 L_{2c} \cos(\theta_1 + \theta_2) \\
 & + m_3 L_1 L_2 \cos(\theta_1 + \theta_2)) \ddot{\theta}_2 + m_3 L_1 L_{3c} \cos(\theta_1 - \theta_3) \ddot{\theta}_3 \\
 & + (-m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) - m_3 L_1 L_2 \sin(\theta_1 + \theta_2) + m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) \\
 & + m_3 L_1 L_2 \sin(\theta_1 + \theta_2)) \dot{\theta}_1 \dot{\theta}_2 + (-m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3) \\
 & + m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3)) \dot{\theta}_1 \dot{\theta}_3 - m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) \dot{\theta}_2^2 \\
 & - m_3 L_1 L_2 \sin(\theta_1 + \theta_2) \dot{\theta}_2^2 + m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3) \dot{\theta}_3^2 \\
 & + m_1 g L_{1c} \cos(\theta_1) + m_2 g L_1 \cos(\theta_1) + m_3 g L_1 \cos(\theta_1)
 \end{aligned} \tag{3.29}$$

The torque for link 2 is:

$$\begin{aligned}
 \tau_2 = & (m_2 L_{2c}^2 + I_2 + m_3 L_2^2) \ddot{\theta}_2 + m_3 L_2 L_{3c} \cos(\theta_3 + \theta_2) \ddot{\theta}_3 \\
 & + (m_2 L_1 L_{2c} \cos(\theta_1 + \theta_2) + m_3 L_1 L_2 \cos(\theta_1 + \theta_2)) \ddot{\theta}_1 \\
 & + (-m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) - m_3 L_1 L_2 \sin(\theta_1 + \theta_2) + m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) \\
 & + m_3 L_1 L_2 \sin(\theta_1 + \theta_2)) \dot{\theta}_1 \dot{\theta}_2 + (-m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2) \\
 & + m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2)) \dot{\theta}_2 \dot{\theta}_3 - m_3 L_1 L_2 \sin(\theta_1 + \theta_2) \dot{\theta}_1^2 \\
 & - m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) \dot{\theta}_1^2 - m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2) \dot{\theta}_3^2 \\
 & + m_2 g L_{2c} \cos(\theta_2) + m_3 g L_2 \cos(\theta_2)
 \end{aligned} \tag{3.30}$$

The torque for link 3 is:

$$\begin{aligned}
\tau_3 = & m_3 L_1 L_{3c} \cos(\theta_1 - \theta_3) \ddot{\theta}_1 + m_3 L_2 L_{3c} \cos(\theta_3 - \theta_2) \ddot{\theta}_2 \\
& + (m_3 L_{3c}^2 + I_3) \ddot{\theta}_3 + (-m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2) + m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2)) \dot{\theta}_2 \dot{\theta}_3 \\
& + (m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3) + m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3)) \dot{\theta}_1 \dot{\theta}_3 \\
& - m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2) \dot{\theta}_2^2 - m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3) \dot{\theta}_1^2 + m_3 g L_{3c} \cos(\theta_3)
\end{aligned} \tag{3.31}$$

The equations for motion shown in Eq. (3.19) for the model, can be rearranged into the following general form shown in Eq. (3.32).

$$D(\theta)\ddot{\theta} + H(\theta, \dot{\theta})\dot{\theta} + G(\theta) = T_\theta \tag{3.32}$$

where $D(\theta)$ represents the inertia matrix, $H(\theta, \dot{\theta})$ represents the matrix of centrifugal and Coriolis forces, $G(\theta)$ represents the matrix of gravitational forces, T_θ represents the vector of external forces and torques applied at the joints. The vectors θ , $\dot{\theta}$, and $\ddot{\theta}$ represent each joints rotational position, velocity and acceleration respectively. In order to reduce the complexity of this model, the backlash of the joints and friction forces are not added. The inertia matrix $D(\theta)$ is shown below, this matrix is a 3×3 symmetric positive definite matrix which defines the kinetic energy of the each link.

$$D(\theta)\ddot{\theta} = \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix} \ddot{\theta} \tag{3.33}$$

The values of the inertia matrix $D(\theta)$ are shown next. From τ_1 in Eq. (3.29), D_{11} , D_{12} , and D_{13} are shown in Eqs. (3.34), (3.35), and (3.36).

$$D_{11} = (m_1 L_{1c}^2 + I_1 + m_2 L_1^2 + m_3 L_1^2) \ddot{\theta}_1 \tag{3.34}$$

$$D_{12} = (m_2 L_1 L_{2c} \cos(\theta_1 + \theta_2) + m_3 L_1 L_{2c} \cos(\theta_1 + \theta_2)) \ddot{\theta}_2 \tag{3.35}$$

$$D_{13} = (m_3 L_1 L_{3c} \cos(\theta_1 - \theta_3)) \ddot{\theta}_3 \quad (3.36)$$

From τ_2 in Eq. (3.30), D_{21} , D_{22} , and D_{213} are shown in Eqs. (3.37), (3.38), and (3.39).

$$D_{21} = (m_2 L_1 L_{2c} \cos(\theta_1 + \theta_2) + m_3 L_1 L_2 \cos(\theta_1 + \theta_2)) \ddot{\theta}_1 \quad (3.37)$$

$$D_{22} = (m_2 L_{2c}^2 + I_2 + m_3 L_2^2) \ddot{\theta}_2 \quad (3.38)$$

$$D_{23} = (m_3 L_2 L_{3c} \cos(\theta_3 + \theta_2)) \ddot{\theta}_3 \quad (3.39)$$

From τ_3 in Eq. (3.31), D_{31} , D_{32} , and D_{33} are shown in Eqs. (3.40), (3.41), and (3.42).

$$D_{31} = (m_3 L_1 L_{3c} \cos(\theta_1 - \theta_3)) \ddot{\theta}_1 \quad (3.40)$$

$$D_{32} = (m_3 L_2 L_{3c} \cos(\theta_3 + \theta_2)) \ddot{\theta}_2 \quad (3.41)$$

$$D_{33} = (m_3 L_2 L_{3c} \cos(\theta_3 + \theta_2)) \ddot{\theta}_3 \quad (3.42)$$

$H(\theta, \dot{\theta})$ matrix groups together Coriolis and centrifugal inertia terms.

$$H(\theta, \dot{\theta}) \dot{\theta} = \begin{bmatrix} 0 & h_{12} & h_{13} \\ h_{21} & 0 & h_{23} \\ h_{31} & h_{32} & 0 \end{bmatrix} \dot{\theta} \quad (3.43)$$

The values of Coriolis and centrifugal inertia matrix $H(\theta, \dot{\theta})$ are shown next.

From τ_1 in Eq. (3.29), h_{12} , and h_{13} are shown in Eqs. (3.44) and (3.45).

$$h_{12} = (-m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) - m_3 L_1 L_2 \sin(\theta_1 + \theta_2) + m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) + m_3 L_1 L_2 \sin(\theta_1 + \theta_2)) \dot{\theta}_1 \dot{\theta}_2 \quad (3.44)$$

$$h_{13} = (-m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3) + m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3)) \dot{\theta}_1 \dot{\theta}_3 \quad (3.45)$$

From τ_2 in Eq. (3.30), h_{21} , and h_{23} are shown in Eqs. (3.46) and (3.47).

$$h_{21} = (-m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) - m_3 L_1 L_2 \sin(\theta_1 + \theta_2) + m_2 L_1 L_{2c} \sin(\theta_1 + \theta_2) + m_3 L_1 L_2 \sin(\theta_1 + \theta_2)) \dot{\theta}_1 \dot{\theta}_2 \quad (3.46)$$

$$h_{23} = (-m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2) + m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2)) \dot{\theta}_2 \dot{\theta}_3 \quad (3.47)$$

From τ_3 in Eq.(3.31), h_{31} , and h_{32} are shown in Eqs. (3.48) and (3.49).

$$h_{31} = (-m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3) + m_3 L_1 L_{3c} \sin(\theta_1 - \theta_3)) \dot{\theta}_3 \dot{\theta}_1 \quad (3.48)$$

$$h_{32} = (-m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2) + m_3 L_2 L_{3c} \sin(\theta_3 + \theta_2)) \dot{\theta}_2 \dot{\theta}_3 \quad (3.49)$$

The gravitational torques matrix $G(\theta)$ Eq. (3.50) represents the terms of gravity for each link of the model.

$$G(\theta) = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \end{bmatrix} \quad (3.50)$$

The values of the gravitational torques matrix $G(\theta)$ Eq. (3.50) are shown in Eqs. (3.51), (3.52), and (3.53):

$$G_1 = m_1 g L_{1c} \cos(\theta_1) + m_2 g L_1 \cos(\theta_1) + m_3 g L_1 \cos(\theta_1) \quad (3.51)$$

$$G_2 = m_2gL_{2c}\cos(\theta_2) + m_3gL_2\cos(\theta_2) \quad (3.52)$$

$$G_3 = m_3gL_{3c}\cos(\theta_3) \quad (3.53)$$

T is the actuating control torque required by each joint during motions, as depicted in Eq. (3.54):

$$T(\theta) = \begin{bmatrix} T_{\theta_1} \\ T_{\theta_2} \\ T_{\theta_3} \end{bmatrix} \quad (3.54)$$

Equation (3.55) shows how Eq. (3.19) is actually related to each link by its angular rotational value. Eq. (3.55) shows the version of Euler general form applied by link 1.

$$D(\theta_1)\ddot{\theta}_1 + H(\theta_1, \dot{\theta}_1)\dot{\theta}_1 + G(\theta_1) = T_{\theta_1} \quad (3.55)$$

where θ_1 shows the link 1 absolute rotational displacement with respect to the horizontal plane. In the matrix form, θ is shown in Eq. (3.58):

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \quad (3.56)$$

In order to model the control properly, the relative angles (q) between each joint need to be used in the model formulations. With this information, the equation of motions is modified as in Eq. (3.57), see Figure 3.4 for all relative angles q between each link. The equation of motions with relative angles is shown in Eq. (3.57):

$$D(q_1)\ddot{q}_1 + H(q_1, \dot{q}_1)\dot{q}_1 + G(q_1) = T_{q_1} \quad (3.57)$$

where q , in its matrix form, is shown as following:

$$q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (3.58)$$

where, q_0 , q_1 , and q_2 are the relative joint angles between links. The relationship between relative angles (q) and absolute angles (θ) are formulated in Eqs. (3.59), (3.60), and (3.61):

$$q_1 = \theta_1 \quad (3.59)$$

$$q_2 = \theta_2 - \theta_1 \quad (3.60)$$

$$q_3 = \theta_3 - \theta_2 - \theta_1 \quad (3.61)$$

The final version of Euler formulation is shown by Eq. (3.62) where a new term called τ_d is added. τ_d represents disturbance applied to the real robot.

$$D(q)\ddot{q} + H(q, \dot{q})\dot{q} + G(q) + \tau_d = \tau \quad (3.62)$$

3.6 Motor and Control Model

The proposed algorithm is based on modifying the gain factor of the motor control system. The motor model used is described in Figure 3.5.

The typical DC Motor model is defined in terms of torque, voltage, and angular velocity according to Eq. 3.5. The variables $\theta(t)$, $\dot{\theta}(t)$ y $\ddot{\theta}(t)$, are position, velocity and angular acceleration respectively. $v(t)$, $i(t)$ y $\tau(t)$, are input voltage, induction current, and output torque. Finally R , L , e , B y J , are induction resistance, electromotive force, friction coefficient, and inertia moment.

Using Kirchoff Voltage Law (KVL) is shown in the Eqs. (3.63) and (3.64):

$$v(t) = Ri(t) + L\frac{di}{dt} + e(t) \quad (3.63)$$

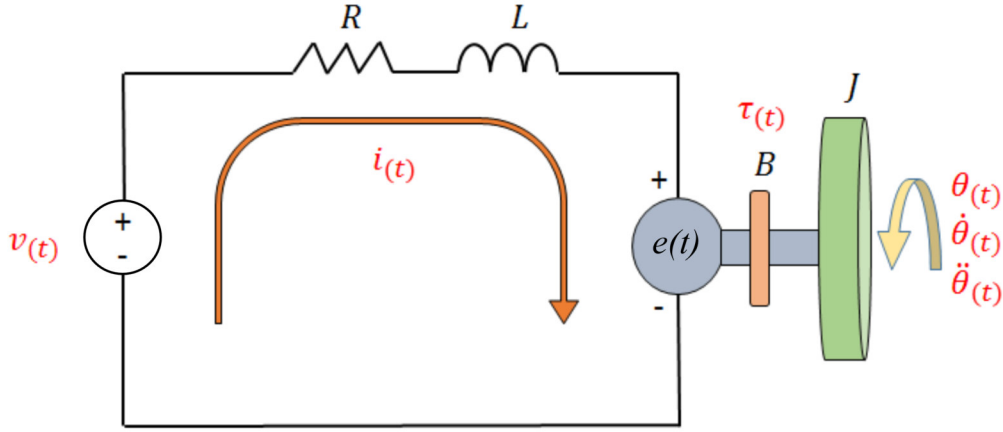


Figure 3.5: Electrical circuit of a typical DC motor.

$$e(t) = K_e \dot{\theta}(t) \quad (3.64)$$

where K_e is the FEM. The electric torque of the motor is defined as shown in Eq. (3.65):

$$\tau(t) = K_t i(t) \quad (3.65)$$

The steady state of Eq. (3.63) is defined in Eq. (3.66):

$$v(t) = Ri(t) + e(t) \quad (3.66)$$

By replacing Eq. (3.64) in Eq. (3.66) and taking $i(t)$ from Eq. (3.67), the following is obtained:

$$i(t) = \frac{v(t)}{R} - \frac{K_e}{R} \dot{\theta}(t) \quad (3.67)$$

Finally, substituting Eq. (3.67) in Eq. (3.65), the torque Eq. (3.68) is obtained:

$$\tau(t) = \frac{K_t}{R} v(t) - \frac{K_e K_t}{R} \dot{\theta}(t) \quad (3.68)$$

Using $K_e = \frac{NRJ}{T_m K_t}$, where T_m , N is the mechanical time and poles number respectively. Applying Laplace transform to Eq. (3.63) results in Eq. (3.69).

$$V(s) = RI(s) + LsI(s) + Ke\dot{\theta}(s) \quad (3.69)$$

Eq. (3.69) can be reorganized as Eq. (3.70):

$$V(s) = I(s)(R + Ls) + Ke\dot{\theta}(s) \quad (3.70)$$

The torque equation shown as a sum of inertial forces and frictions results in Eq. (3.71):

$$\tau(t) = J\ddot{\theta}(t) + B\dot{\theta}(t) \quad (3.71)$$

Replacing Eq. (3.65) in Eq. (3.71) gives an expression in terms of $i(t)$ and $\dot{\theta}(t)$, resulting in Eqs. (3.72) and (3.73) by applying the Laplace transform:

$$K_t i(t) = J \frac{d\dot{\theta}(t)}{dt} + B\dot{\theta}(t) \quad (3.72)$$

$$K_t I(s) = \dot{\theta}(s)(Js + B) \quad (3.73)$$

From Eq. (3.73), $I(s)$ is obtained in Eq. (3.74) and $V(s)$ of Eq. (3.75) is obtained from Eq. (3.70):

$$I(s) = \dot{\theta}(s) \left(\frac{Js}{K_t} + \frac{B}{K_t} \right) \quad (3.74)$$

$$V(s) = \dot{\theta}(s) \left(\left(\frac{Js}{K_t} + \frac{B}{K_t} \right) (R + Ls) + Ke \right) \quad (3.75)$$

Then, using Eq. (3.75) and expressing the input in terms of $V(s)$ and output as $\dot{\theta}(s)$, results in the transfer function $G(s)$ described in Eq. (3.76).

$$G(s) = \frac{\dot{\theta}(s)}{V(s)} = \frac{K_t}{s^2 JL + s(RJ + BL) + BR + Ke K_t} \quad (3.76)$$

From Eq. (3.76) it is possible to consider: B is small as $B \approx 0$, then $RJ \gg BL$ and $K_e K_t \gg BR$. Then the last expression of the transfer function $G(s)$ is obtained as shown by Eq. (3.77):

$$G(s) = \frac{K_t}{JLs^2 + RJ s + K_e K_t} \quad (3.77)$$

Multiplying $(\frac{R}{K_e K_t}) (\frac{1}{R})$ by (3.77), results in Eq. (3.78).

$$G(s) = \frac{\frac{1}{K_e}}{(\frac{RJ}{K_e K_t})(\frac{L}{R})s^2 + (\frac{RJ}{K_e K_t})s + 1} \quad (3.78)$$

Finally, if $T_m = \frac{RJ}{K_e K_t}$ and $T_e = \frac{L}{R}$, the transfer function is depicted by Eq. (3.79).

$$G(s) = \frac{\frac{1}{K_e}}{T_m T_e s^2 + T_m s + 1} \quad (3.79)$$

Next, an example about the effect of reducing the proportional gain of the DC motor PID controller is depicted. This simple experiment tries to illustrate the idea proposed in this work, where a low stiffness is reached changing the proportional gain value of the PD or PID controller. The mathematical model used for the DC motor is described by Eq. (3.77). The control motor system for current example is a simple PD control as depicted in Figure 3.6.

Figure 3.6 shows the PD motor control scheme, where K_p and K_d are the position and velocity gains respectively, $\dot{\theta}_d$ and θ_d are the desired velocity and position. The proposed work is based on the generation of variable stiffness through the manipulation of K_p in the feedback loop. The K_p gain affects the control signal as shown by Eq. (3.80).

$$u(t) = -k_d \dot{e}(t) - k_p e(t) \quad (3.80)$$

When K_p is small, the control signal is weak and the system is easily displaced from the target position. On the other hand, when K_p is large, the system tries to keep the target position using all the energy provided by the control signal. Figure 3.7 shows the effect of varying the K_p value, showing the position displace-

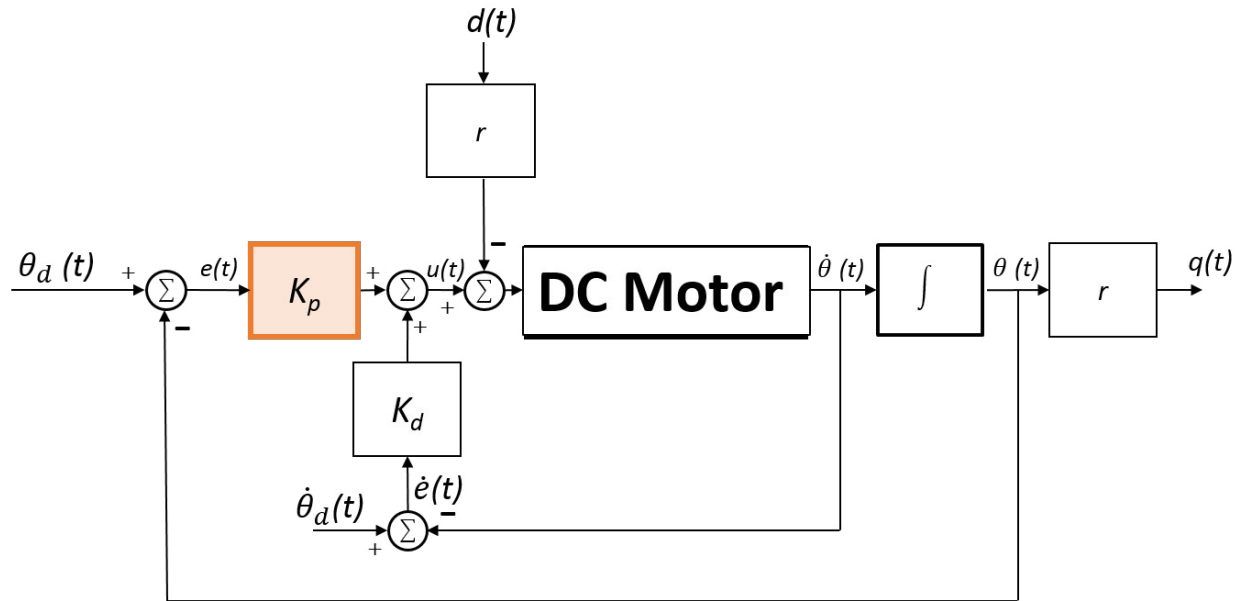


Figure 3.6: PD control model for DC motor

ment for different values of the constant over a simulation of the control scheme. During the simulation, a disturbance signal is applied at time $t = 10$.

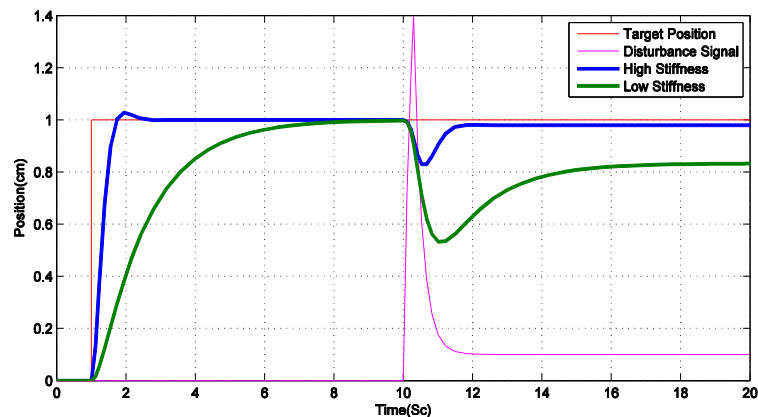


Figure 3.7: Step response of motor model with low and high stiffness

Figure 3.7 shows the step response of the motor model using two different values of K_p . The model control system is exposed to a disturbance input $d(t)$. This is shown as a magenta line in Figure 3.7. The blue line is the response using a high K_p gain and the green one is the response using a low K_p gain. With

high K_p values the system is less affected by disturbance input, on the other hand, when the K_p value is low, the disturbance affects considerably the motor position.

The application of low values in the proportional gain for the previous motor model produces the position displacement from the target position according to with the gain variation. Generally, the motor displacement from the target position is an undesired effect, however in this work, the displacement from target position helps to reduce the impact force in the robot falling event and the landing phase in jumping movements.

CHAPTER 4

LOW STIFFNESS GENERATION AND JUMPING CONTROL

4.1 Note to the Reader

Part of this chapter was published in the IEEE-SoutheastCon [26] and the proceedings of following conference: Innovations in Bio-Inspired Computing and Applications 2015 [27].

4.2 Introduction

This chapter introduces the control system employed to perform vertical jumps with humanoid robots, a low stiffness generation method based on optimal control approaches.

4.3 Control of the Take-off Phase

This section presents the necessary conditions to perform the take-off phase of vertical jump and the control strategy to ensure the required trajectories to achieve accelerations of the robot CoM over the gravity.

4.3.1 Vertical Jump Conditions

Using a similar approach from Babič et al. in [46], two primary conditions to achieve a vertical jump are considered. The first one, the CoM has to move upward and the displacement from the point zero in the horizontal axes has to be minimal; also, the CoM must stay inside the support polygon, i.e. the foot of the robot. The second one is related to ZMP [24], where the components in the horizontal axes have to be equal to zero.

The CoM position is defined by Eq. (4.1), where x_{com} and y_{com} are the horizontal and vertical position of the robot's CoM, m_i is the mass of the i -th link, x_i and y_i are the coordinates of the CoM of the i -th link.

$$x_{com} = \frac{\sum_{i=1}^n m_i x_i}{\sum_{i=1}^n m_i}, y_{com} = \frac{\sum_{i=1}^n m_i y_i}{\sum_{i=1}^n m_i} \quad (4.1)$$

The ZMP is defined by Eqs. (4.2) and (4.3), where ω_i is the angular velocity of the i -th link, and \mathbf{I}_i is the inertial tensor of the i -th link around the CoM.

$$x_{zmp} = \frac{\sum_{i=1}^n m_i x_i (\ddot{y}_i + g) - \sum_{i=1}^n m_i y_i \ddot{x}_i + \tau_z}{\sum_{i=1}^n m_i (\ddot{y}_i + g)} \quad (4.2)$$

$$\tau_z = \sum_{i=1}^n (\mathbf{I}_i \dot{\omega}_i + \omega_i \times \mathbf{I}_i \omega_i) \quad (4.3)$$

Using Eq. (4.1), the second derivative of x_{com} and y_{com} can be formulated for control purposes, and x_{zmp} can be computed as shown in the Eqs. (4.4) to (4.6):

$$\ddot{x}_{com} = \alpha_1 \ddot{q}_1 + \alpha_2 \ddot{q}_2 + \alpha_3 \ddot{q}_3 + d_1 \quad (4.4)$$

$$\ddot{y}_{com} = \beta_1 \ddot{q}_1 + \beta_2 \ddot{q}_2 + \beta_3 \ddot{q}_3 + d_2 \quad (4.5)$$

$$x_{zmp} = \gamma_1 \ddot{q}_1 + \gamma_2 \ddot{q}_2 + \gamma_3 \ddot{q}_3 + d_3 \quad (4.6)$$

where α_i , β_i , γ_i , and d_i are functions of joint angles (q_i). Finally, in Eq. (4.7) the acceleration of the desired trajectory (\ddot{q}_d) can be expressed in terms of α_i , β_i , γ_i , and d_i using Eqs. (4.4), (4.5), and (4.6)

$$\ddot{q}_d = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix}^{-1} \left(\begin{bmatrix} \ddot{x}_{com} \\ \ddot{y}_{com} \\ 0 \end{bmatrix} - \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} \right) \quad (4.7)$$

where

$$\ddot{q}_d = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} \quad (4.8)$$

4.3.2 Computed-Torque Control

The trajectories to ensure the detachment from the ground were defined in the last section. Those trajectories are necessary to achieve a CoM acceleration greater than gravity. To follow those trajectories, it is crucial to apply a control strategy to guarantee that the robot takes off from the ground. For this reason, Computed-Torque Control is used as a control strategy to track trajectories necessary in the takeoff phase.

Computed-Torque Control is a widely used control strategy based on two special approaches [58], [59]. The first one uses feedback linearization of nonlinear systems. The second one is based on a computation of the robot's required torque by the use of the nonlinear feedback control law [60]. This kind of control is based on the concept that there is a desired tracking signal and the system tries to follow it. The main idea here is to reduce the error through a feedback linearization of the system.

The error is defined as the difference between the desired trajectory and the actual joint position as shown in Eq. (4.9). The variables $\dot{e}(t)$ and $\ddot{e}(t)$ can be defined using a similar approach.

$$e(t) = q_d(t) - q(t) \quad (4.9)$$

$$\dot{e}(t) = \dot{q}_d(t) - \dot{q}(t) \quad (4.10)$$

$$\ddot{e}(t) = \ddot{q}_d(t) - \ddot{q}(t)$$

where $q(t)$ is the current position of the actuator Eq. (4.12), and it is defined from dynamic robot model deduced in section 3.5 given by Eq. (4.11).

$$D(q)\ddot{q}(t) + H(q, \dot{q})\dot{q}(t) + G(q) + \tau_d(t) = \tau(t) \quad (4.11)$$

From Eq. (4.11) $\ddot{q}(t)$ can be derived as shown by Eq. (4.12).

$$\ddot{q}(t) = D^{-1}(q)(H(q, \dot{q})\dot{q}(t) + G(q) + \tau_d(t) - \tau(t)) \quad (4.12)$$

Now by back substitution of Eq. (4.12) into Eq. (4.9), the second derivative of error is obtained as shown in Eq. (4.13).

$$\ddot{e}(t) = \ddot{q}_d(t) + D^{-1}(q)(H(q, \dot{q})\dot{q}(t) + G(q) + \tau_d(t) - \tau(t)) \quad (4.13)$$

The variable $u(t)$ is defined as the control input function, and $w(t)$ as the disturbance function, as shown below.

$$u(t) = \ddot{q}_d(t) + D^{-1}(q)(H(q, \dot{q})\dot{q}(t) + G(q) - \tau(t)) \quad (4.14)$$

$$w(t) = D^{-1}(q)\tau_d(t) \quad (4.15)$$

The feedback linearization of Eq. (4.14) can be inverted to yield τ as given in Eq. (4.16).

$$\tau(t) = D(\ddot{q}_d(t) - u(t)) + H(q, \dot{q})\dot{q}(t) + G(q) \quad (4.16)$$

where $u(t)$ is the control signal and it can be selected as P, PD or PID feedback loop control signal.

4.3.3 PID Outer Loop

This research work proposes the use of PID Computed-Torque Controller. The PID design starts including an integrator in the feed-forward loop.

$$\dot{\varepsilon}(t) = e(t) \quad (4.17)$$

$$u(t) = -k_d\dot{\varepsilon}(t) - k_p e(t) - k_i \varepsilon(t) \quad (4.18)$$

To conclude, Eq. (4.18) is back substituted into Eq. (4.16), which yields the final form for $\tau(t)$ as shown in Eq. (4.19). Figure 4.1 depicts the Computed-Torque Control schema with the outer PID loop feedback.

$$\tau(t) = D(\ddot{q}_d(t) + k_d\dot{e}(t) + k_p e(t) + k_i \varepsilon(t)) + H(q, \dot{q})\dot{q}(t) + G(q) \quad (4.19)$$

where $\varepsilon(t)$ is the integral of the tracking error $e(t)$. Thus additional dynamics have been added to the linear outer loop compensator.

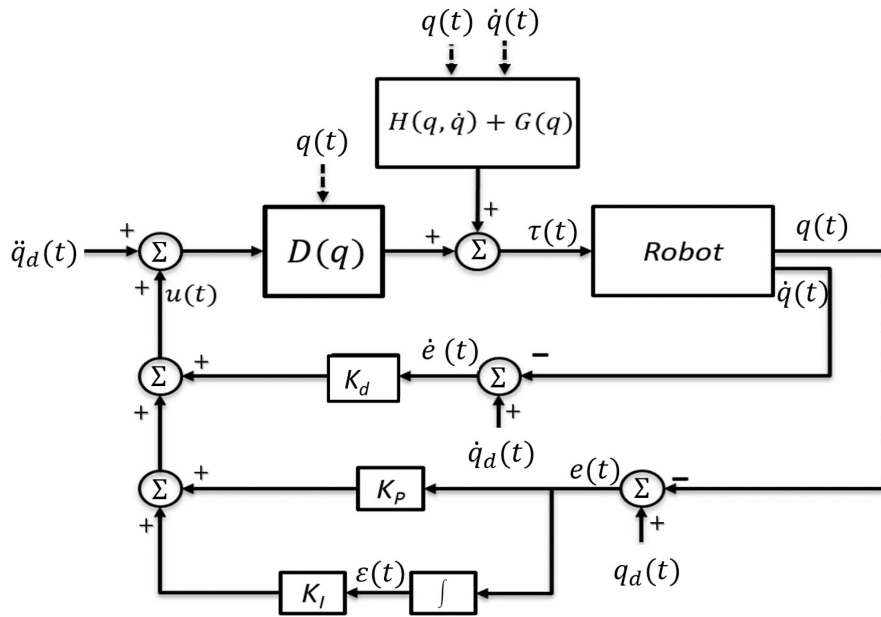


Figure 4.1: Computed-Torque Control schema with outer PID loop

The Computed-Torque Control is applied during all four jumping phases. The system tracks the desired trajectory $(\ddot{q}_d(t), \dot{q}_d(t), q_d(t))$ on the take-off phase. For the flight and landing phases, the control system tries to keep the stand-up position and reject the disturbance produced by the ground impact.

4.4 Landing Phase

One of the principal aims of this work is the generation of soft landing during a vertical jumping movement of a real robot. The approach used to accomplish this objective is the use of variable stiffness in the ankle and knee joints. The variable stiffness is used to reach values where low stiffness is obtained through

small changes of the control parameters. Low stiffness values allow for the absorption of impact force in the landing phase of the jumping movement.

The current work proposes two different approaches to generating low stiffness. The principal approach uses an optimal control approach based on Linear Quadratic Regulator (LQR). In this research, this method is labeled Optimal Variable Stiffness. The second approach is inspired by the Fuzzy logic theory and it will be depicted in chapter 6 of this work.

4.4.1 Low Stiffness Generation

The proposed main approach used to generate variable stiffness is based on optimal control theory. The variable stiffness is reached using Linear Quadratic Regulator (LQR) calculations. In general terms, this design is based on the LQR calculation when the DC motor is working with full stiffness (Control feedback values for normal performance). Once the control feedback values for the motor normal performance has been calculated, the weighted matrix (Q) between energy consumption and position error is premultiplied by the ρ factor between 0 and 1. The aim of the ρ factor is to set up a different relation between input energy and position error. For ρ values below of 1, the position error is increased and stiffness is reduced. Then the new Q matrix is calculated using the ρ factor. The new Q matrix is then used to redesign for the LQR to generate low stiffness in order to achieve the complete system performance, i.e. impact force reduction. The ρ parameter determines the stiffness level for DC Motor control.

As mentioned in the last section, this work bases the impact reduction strategy in the motor stiffness variation using an LQR approach. Thus, this section is divided into three parts. The first one is the mathematical model of the electrical motor, stating the restrictions of the used model with respect to the general one presented above. The second one is the formulation of the optimal control approach, and third the calculation of the PID constants from the LQR, according to the requirements of the robotic platform dynamics.

4.4.1.1 Electrical Motor Model

Since most of the actual robot actuators are electric motors, the present work uses the mathematical model of a DC electric motor for the controller design. The transfer function for a DC Motor is presented by Eq. (4.20)

$$\frac{\omega(s)}{V_{in}(s)} = \frac{K_t}{(Js + b)(Ls + R) + K_t K_b} \quad (4.20)$$

where, $V_{in}(s)$ is the input voltage, $\omega(s)$ is angular velocity, L is the armature inductance, R is the armature resistance, b is the motor viscous friction constant, J is the moment of the inertia of the rotor, K_t is the motor torque constant, and K_b is the electromotive force constant.

The motor transfer function is depicted by Eq. (4.20) as the relation between velocity and input voltage. This model can be extended to a position model adding an integrator and gear ratio (Gr) to the output. The new function model looks as shown by Eq. (4.21).

$$\frac{\theta(s)}{V_{in}(s)} = \frac{K_t Gr}{s((Js + b)(Ls + R) + K_t K_b)} \quad (4.21)$$

Finally, the state space representation is depicted by Eq. (4.22).

$$\frac{d}{dt} \begin{bmatrix} \Theta \\ \dot{\Theta} \\ i \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{b}{J} & \frac{K}{J} \\ 0 & -\frac{K}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \Theta \\ \dot{\Theta} \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} V \quad (4.22)$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Theta \\ \dot{\Theta} \\ i \end{bmatrix} \quad (4.23)$$

4.4.1.2 Linear Quadratic Regulator

The Linear Quadratic Regulator (LQR) based optimal control has been widely studied over decades with a broad range of applications. It minimizes the error in the state variable trajectories of a system while requiring the minimum control energy. The objective to use LQR is to generate a variable stiffness effect in the actuator, DC electric motor, as it will be explained later. LQR is based on the minimization of the performance index, J , as shown by Eq. (4.24) below.

$$J = \int_0^{\infty} \left[x^T(t)Qx(t) + u^T(t)Ru(t) \right] dt \quad (4.24)$$

where Q and R are the penalization matrices for state variables error and control signal, respectively. The relation between Q and R determines what is more important between the minimization of state error or the control energy. Thereby, the present work proposes to determine the stiffness of the motors using the Q matrix. When Q has large values, the motor position error is penalized. Then, the motor tries to keep the position against any disturbance, high stiffness. Otherwise, if Q has small values, a certain position error is allowed, generating low stiffness in the motor. The feedback control law is defined by Eq. (4.25).

$$u(t) = -R^{-1}B^T Px(t) = -Kx(t) \quad (4.25)$$

where, K is the Kalman gain and it is defined as $K = R^{-1}BP$ and P is a symmetric positive definite matrix and it is a solution of the Continuous Algebraic Riccati equation defined by Eq. (4.26).

$$A^T P + PA - PBR^{-1}P + Q = 0 \quad (4.26)$$

here, A and B are the matrices of the state space description of the plant, i.e. motor, Q is symmetric positive semi-definite weighted matrix, and R is a positive constant diagonal matrix. Using the Riccati equation, the feedback gains can be calculated and the design of the control system can be performed. However, the system performance depends on the adequate selection of the Q matrix. The next section explains how to calculate Q based on the desired model and how to tune a PID using the LQR approach.

4.4.1.3 Optimal PID Design

PID is a control system widely used around the world. This work tries to provide a method to vary the stiffness in a motor using PID control, using an optimal approach. The variable stiffness is used to reduce the impact force in falling robots, as previously mentioned. The design of optimal PID controller is based in the works presented by [61] and [62]. They first proposed a PID design using an LQR approach, however the selection of the Q matrix was not defined. The second one design the PID using LQR and provides

a method to determine the Q values, based on the characteristic polynomial of the desired behavior. The current work assumes the design of the PID controller as an LQR optimal control design, where the error is described as a function of state variables and the optimal state feedback gains are the PID parameters (K_p , K_i , and K_d). Figure 4.2 shows the typical PID configuration, where $r(t)$ is the desired actuator position.

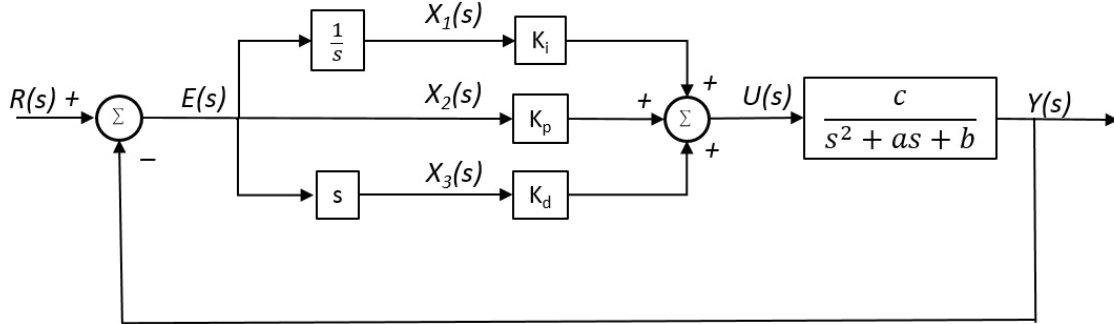


Figure 4.2: PID controller for second order system

The PID transfer function and the second order plant (motor) are defined by Eq. (4.28) and Eq. (4.27) respectively.

$$U(s) = E(s) \left(\frac{K_i}{s} + K_p + K_d s \right) \quad (4.27)$$

$$G(s) = \frac{c}{s^2 + as + b} = \frac{Y(s)}{U(s)} \quad (4.28)$$

Now, the state variables are defined in Eq. (4.29).

$$x_1(t) = \int e(t)dt, \quad x_2(t) = e(t), \quad x_3(t) = \frac{de(t)}{dt} \quad (4.29)$$

For the feedback design, the external desired set-point does not affect the controller design and it is possible to assume $r(t) = 0$, thus $e(t) = -y(t)$. This is a common assumption in the standard regulator design. Assuming $r(t) = 0$ the transfer function can be expressed as shown in Eq. (4.30).

$$\frac{Y(s)}{U(s)} = \frac{c}{s^2 + as + b} = \frac{-E(s)}{U(s)} \quad (4.30)$$

Now the relation between $U(s)$ and $E(s)$ is written in the time domain and shown in Eq. (4.31).

$$\ddot{e}(t) + a\dot{e}(t) + be(t) = -cu(t) \quad (4.31)$$

replacing Eq. (4.30) in Eq. (4.31) the relation between $u(t)$ and $y(t)$ is expressed in terms of the state variables as depicted in Eq. (4.32).

$$\dot{x}_3(t) + ax_3(t) + bx_2(t) = -cu(t) \quad (4.32)$$

Finally, using Eqs. (4.29) and (4.32) the state space formulation is depicted in Eq. (4.33).

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -b & -a \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -c \end{bmatrix} u(t); \quad A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -b & -a \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ -c \end{bmatrix} \quad (4.33)$$

Now it looks as a standard state-space representation $\dot{x}(t) = Ax(t) + Bu(t)$, where A and B are shown in Eq. (4.33).

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -b & -a \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ -c \end{bmatrix} \quad (4.34)$$

Then, using Eq. (4.26) the Riccati equation Solution can be applied using A and B from Eq. (4.33) and P is defined as a 3x3 symmetric matrix as depicted in Eq. (4.35).

$$P = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix} \quad (4.35)$$

Using the results of P matrix, the K gains are obtained using Eq. (4.25) where $K_1 = K_i$, $K_2 = K_p$, and $K_3 = K_d$.

4.4.1.4 Optimal Variable Stiffness Design Using Optimal PID

In order to develop variable stiffness in a motor using an optimal PID design, it is necessary to establish the desired transfer function according to the desired performance. Usually, the desired function is determined according to requirements of settling time, overshoot and rise time, among others. However, for the present case, those parameters can be defined according to the normal performance of the motor in common robot activities like walking.

The idea is to set up an operating point where the motor stiffness will be defined as a normal stiffness. Starting from this point a low and high stiffness will be defined. Thus, the desired function is defined as a third order system, where the characteristic polynomial has three roots labeled by $-\alpha_1$, $-\alpha_2$, and $-\alpha_3$.

The objective of this section is to design a PID controller, starting from the desired performance using an optimal approach. The PID transfer function is defined as shown in Eq. (4.36).

$$G_{PID}(s) = k_p + \frac{k_i}{s} + k_d s \quad (4.36)$$

Following the method described by [62], Q and P matrices are defined as shown in Eqs. (4.37) and (4.35), respectively.

$$Q = \begin{bmatrix} q_1 & 0 & 0 \\ 0 & q_2 & 0 \\ 0 & 0 & q_3 \end{bmatrix} \quad (4.37)$$

According to the optimal conditions, where P has to be a solution of the Riccati Eq. (4.26). The values of Q can be set in terms of the motor transfer function coefficients and the roots of the desired polynomial characteristic equation. The Q values are depicted in Eqs. (4.38)-(4.40).

$$q_1 = \frac{R\alpha_1^2\alpha_2^2\alpha_3^2}{c^2} \quad (4.38)$$

$$q_2 = \frac{R(\alpha_1^2\alpha_2^2 + \alpha_1^2\alpha_3^2 + \alpha_2^2\alpha_3^2 - b^2)}{c^2} \quad (4.39)$$

$$q_3 = \frac{R(\alpha_1^2 + \alpha_2^2 + \alpha_3^2 - a^2 + b^2)}{c^2} \quad (4.40)$$

Once Q matrix has been defined, the Riccati equation is applied using Eq. (4.26) and then P matrix is obtained. The solution of the Riccati equation can be found using numerical methods or mathematical software, such as Matlab. Now, the optimal PID constants are computed using Eq. (4.41).

$$K_i = R^{-1}cP_{13}; \quad K_p = R^{-1}cP_{23}; \quad K_d = R^{-1}cP_{33} \quad (4.41)$$

Because the optimal PID gains were calculated using a characteristic polynomial of the desired transfer function with the usual performance parameters, the stiffness in the motor is assumed as the usual stiffness. However, based on this design, it is possible to calculate a new set of optimal PID gains. These new gains can produce a high or low stiffness from the usual stiffness perspective. The low and high stiffness can be designed scaling the Q_{usual} matrix by a ρ factor as shown by Eq. (4.42).

$$Q_{new} = \rho Q_{usual} \quad (4.42)$$

where ρ is a scalar parameter, for $\rho > 1$ a high stiffness is obtained and $0 < \rho < 1$ the low stiffness will be reached by the motor. Finally, with the Q_{new} defined, it is necessary to solve the Riccati equation for Q_{new} and get the P values to calculate the new PID control gains.

4.4.2 Control of Landing Phase

The requirements to perform a vertical jump with a One-legged robot, up to this point, have been implemented and tested from the take-off to the landing stages, including the stiffness reduction to preserve the DC motor integrity for the particular joint.

The stiffness reduction is based on the trade-off between position error and energy consumption. The low stiffness causes the position error to increase. This error generates the displacement of the robots CoM with respect to the point zero. When the CoM goes out of the contact support area, usually the feet area, the robot falls down.

The main objective of this section is to present the design of the control system to keep the robots CoM close to the point zero within the support area. This system is called the landing control strategy. The landing control strategy has as the aim to preserve the CoM close to the zero point as explained previously.

To achieve this objective, the control strategy is composed of different computational function blocks such as body inclination, forward kinematic, calculation and correction of the CoM, delta ρ calculation, PID controller with Computed-Torque Control. The landing control strategy is depicted by Figure 4.3 and each unit of the control system is explained in the next paragraphs.

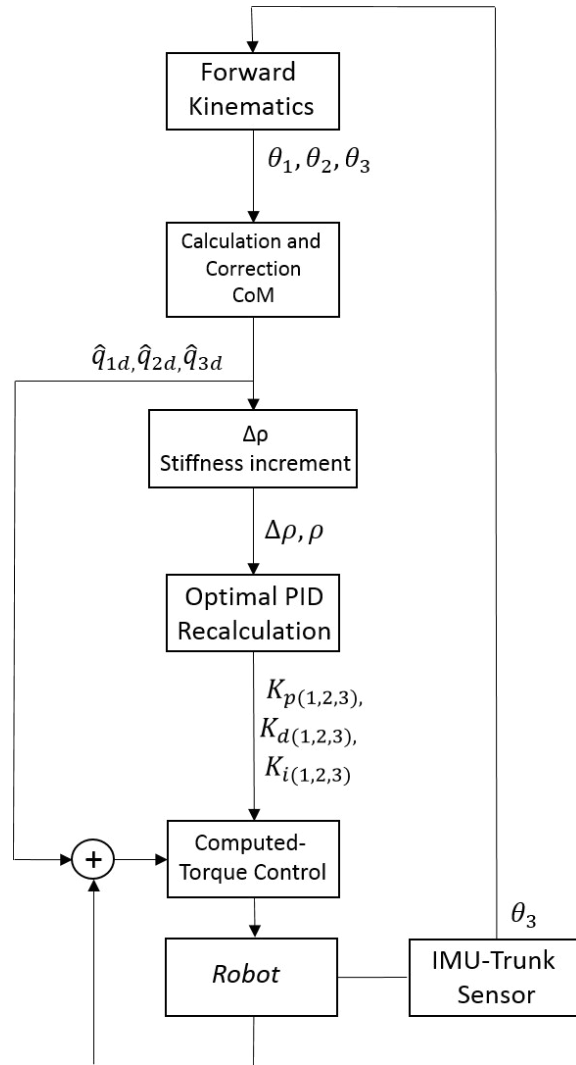


Figure 4.3: General schema control landing strategy

4.4.2.1 Body Inclination Sensor

In order to keep the balance in the robot, it is necessary to have information about the trunk inclination, since sometimes in the landing moment, the feet dont make full contact with the ground as shown in Figure

4.4. Figure 4.4 shows the moment when the robot is touching the ground for the first time and the contact is done just in one corner of the foot. This is a problem because in order to calculate the CoM one must

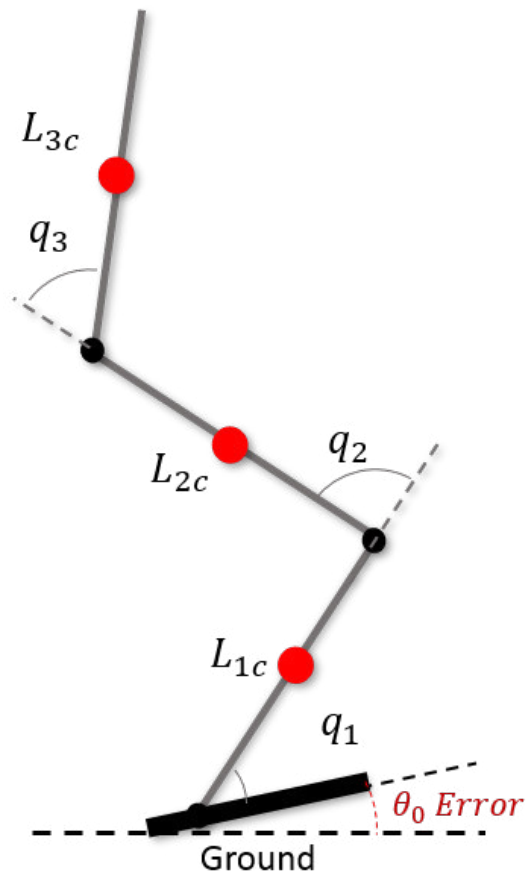


Figure 4.4: Foot angle error in landing phase

know the absolute angle of each link θ_i , which is not possible if the foot is not parallel to the ground. This problem is produced by small rotations of the CoM in the flight phase. This small rotation is usually a consequence of a poor trajectory tracking done by the motors in the take-off phase. Thus, it is necessary to calculate θ_i for every link using information from the robots sensors. Usually, humanoid robots have an Inertial Measurement Unit (IMU) embedded into the body, which is usually located in the trunk. The information that is generated at the IMU is processed by a Kalman Filter. The outcome of the filter is θ_3 , which is the angle between the trunk and horizontal axis, i.e. the absolute inclination of the trunk.

4.4.2.2 Forward Kinematics

This functional block is in charge of computing θ_2 and θ_1 . These calculations are done using the values of θ_3 , q_3 , q_2 , and q_1 , where θ_3 is provided by the IMU and q_i is the relative angles between links. The q_i values are obtained through the use of rotational encoders in every joint. The values of θ_1 and θ_2 are calculated using Eqs. (4.43) and (4.44).

$$\theta_2 = \theta_3 - q_3 \quad (4.43)$$

$$\theta_1 = \theta_2 - q_2 \quad (4.44)$$

4.4.2.3 Calculation and Correction of Center of Mass

It is important to keep the CoM position close to the point zero to prevent the robot from falling when in the landing phase. The CoM is defined by Eq. (4.45) as a summation of the CoM of each link multiplied by the mass of each link and divided by total mass. Figure 4.5 shows the CoM for every link in red circles, total robot CoM in the blue circle and the position of the point zero in the foot support in the green circle.

$$x_{com} = \frac{\sum_{i=1}^n m_i x_i}{\sum_{i=1}^n m_i}, \quad y_{com} = \frac{\sum_{i=1}^n m_i y_i}{\sum_{i=1}^n m_i} \quad (4.45)$$

The CoM is depicted by Eq. (4.45), where i goes from 1 to 3 according to the number of robot links. Then, x_{com} and y_{com} can be expressed as shown by Eqs. (4.46) and (4.47).

$$x_{com} = \frac{m_1 x_1 + m_2 x_2 + m_3 x_3}{m_1 + m_2 + m_3} \quad (4.46)$$

$$y_{com} = \frac{m_1 y_1 + m_2 y_2 + m_3 y_3}{m_1 + m_2 + m_3} \quad (4.47)$$

where x_1 , x_2 , and x_3 are the global position of CoM for every link. L_{ic} are the positions of the CoM of each link. Eqs. (4.48), (4.49), and (4.50) depict the global position in X axis of the CoM for each link.

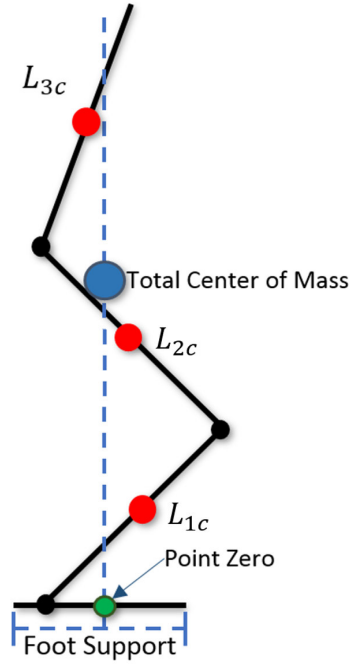


Figure 4.5: CoM and zero point

$$x_1 = L_{1c} \cos(\theta_1) \quad (4.48)$$

$$x_2 = L_1 \cos(\theta_1) + L_{2c} \cos(\theta_2) \quad (4.49)$$

$$x_3 = L_1 \cos(\theta_1) + L_2 \cos(\theta_2) + L_{3c} \cos(\theta_3) \quad (4.50)$$

where L_i is the longitude of each link, L_{ci} is the relative position of CoM of each link, and θ_i is the angle between each link and the X axis. The relative angle between each link is denoted by q_i , where the relation of θ_i and q_i is depicted by Eqs. (4.51), (4.52), and (4.53).

$$\theta_1 = q_1 \quad (4.51)$$

$$\theta_2 = q_1 + q_2 \quad (4.52)$$

$$\theta_3 = q_1 + q_2 + q_3 \quad (4.53)$$

The correction of the CoM is performed through position adjustment of q_2 and q_3 . The correction of q_2 is performed by equating to zero the sum of the CoM of x_1 and x_2 as shown by Eq. (4.54)

$$L_{c1}\cos(\theta_1) + L_1\cos(\theta_1) + L_{2c}\cos(\hat{\theta}_{2d}) = 0 \quad (4.54)$$

where $\hat{\theta}_{2d}$ is the position of the joint required to accomplish Eq. (4.54). Likewise, the correction of q_3 is calculated doing the sum of x_1 , x_2 , and x_3 then equating it to zero, as shown by Eq. (4.55).

$$L_{c1}\cos(\theta_1) + L_1\cos(\theta_1) + L_{2c}\cos(\theta_2) + L_1\cos(\theta_1) + L_2\cos(\theta_2) + L_{3c}\cos(\hat{\theta}_{3d}) = 0 \quad (4.55)$$

where $\hat{\theta}_{3d}$ is the position required for the trunk to solve the Eq. (4.55). Using these values, then the relative angles between links are calculated. Eqs. (4.56) and (4.57) show the relationship between the current angles and the desired angles.

$$\hat{q}_{2d} = \hat{\theta}_{2d} - \theta_1 \quad (4.56)$$

$$\hat{q}_{3d} = \hat{\theta}_{3d} - \theta_2 \quad (4.57)$$

where \hat{q}_{3d} and \hat{q}_{2d} are the angles required by joints 2 and 3 to make the robot CoM go to zero.

4.4.2.4 ρ Adjustment Calculation

This unit computes the rate of change for the stiffness factor defined as ρ in the section 4.4.1.4. When the robot is falling, the stiffness is decreased to absorb the impact force. However once the robot has landed, is necessary to keep the balance and set the CoM close to the zero point.

In order to perform the position correction of the links, the joint stiffness must be increased again. The stiffness factor should slowly return to its maximum value equal to one. To do this, $\Delta\rho$ is defined as the

correction factor as shown in Eq. (4.58).

$$\Delta\rho = \alpha(q - \hat{q}_d)/q_d \quad (4.58)$$

where α is the proportional factor according to the desired correction. Usually $0 \leq \alpha < 1$. With $\alpha = 1$, the correction can be strong and with α close to 0, the correction could be soft and slow. This correction depends on the speed required to compensate the CoM. The final value of ρ is defined by $\rho_{new} = \rho_{current} + \Delta\rho$. In practical terms, $\Delta\rho$ is between 0 and 0.2; however it will depend on how often ρ is updated.

4.4.2.5 Optimal PID Recalculation

This functional block makes the calculation of the PID gains for every joint according to section 4.4.1.4, using the updated values of ρ . The final control system is composed by the ρ adjustment and the Computed-Torque Control for the new desired position (\hat{q}_{id}) as shown by Figure 4.6.

Figure 4.6 shows the complete schema of the landing control system strategy. Every stage of the control system is depicted including the Computed-Torque Control. The Computed-Torque Control is actuating over the robot and dealing with the non-linear effect of the inertial and friction components of the robot. Figure 4.6 shows the inputs and outcomes of every stage. Next section shows simulations of the complete control landing system strategy.

4.4.3 Simulation of the Control Landing System Strategy

The performance of the jumping movement depends on the absorption of the impact force and on maintaining balance in the landing phase.

The landing control system strategy was designed taking into account the impact absorption and balance of the robot. The simulation focuses on the study of the CoM trajectory in the landing phase. In order to observe the CoM trajectory, joint trajectories for the knee, ankle, and hip are previously studied.

Figure 4.7 shows the trajectory of each joint, ankle, knee, and hip, with low stiffness conditions without control feedback and error correction. This figure depicts the robot behavior without applying the landing control system strategy, i.e. open loop control. The blue line shows the trajectory of the link with low stiffness

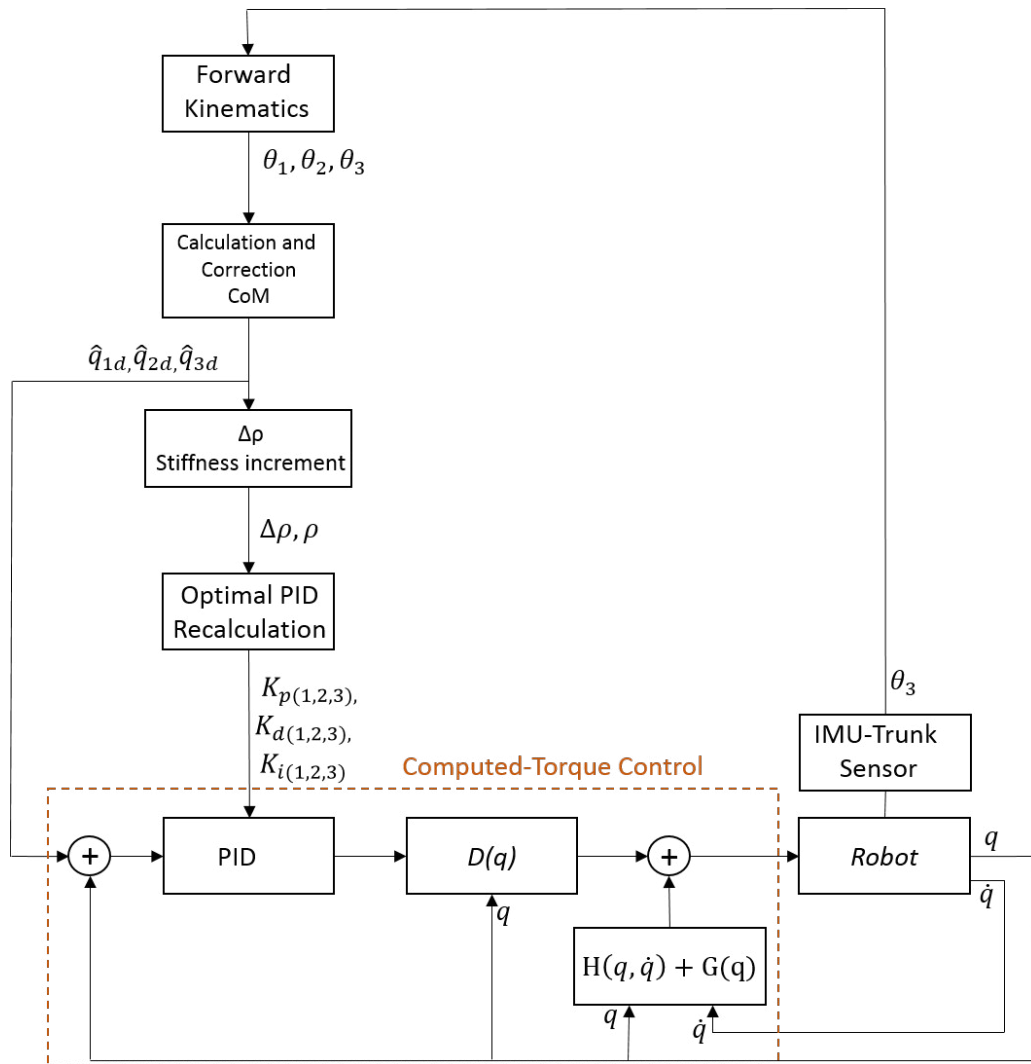


Figure 4.6: Complete schema control landing

after the impact moment with the dashed orange line as the desired trajectory. The figure shows how the joint position is displaced from the desired position. The desired position is set up previous to the landing phase and it is the value necessary to keep the CoM close to the zero point.

Figure 4.8 shows the joint positions for the ankle, knee, and hip joints. This case shows the performance of the stiffness control system strategy during the landing phase. One should take into account the low stiffness of every joint at beginning of the landing stage. Once the robot makes contact with the ground, the impact force produces a displacement of every joint from the desired position. The landing control system tries to reduce the difference between the real joint position and desired position during the landing phase.

The blue line is the current position of the joint after the impact and the orange dashed line is the recalculated desired position for the joint. The desired position is recalculated by the "Calculation and correction of CoM" functional block. This part of the process is necessary because the low stiffness allows for the increment of the error between the joint position and the pre-calculated desired position. It is necessary to recalculate the desired positions since the total error is a combination of the errors of every joint. Because of it, the new desired position is calculated using Eqs. (4.54) and (4.55).

Figure 4.8 shows how these desired positions are varying through time. Also, at the end of this phase, the position of the joints reach the desired position, due to the control system is actuating and reducing each link error between its current and desired positions.

Figure 4.9 shows a comparison between the desired trajectory, dashed yellow line, for each joint, the current joint position with stiffness control strategy, blue line, and the joint position for low stiffness without control position adjustment, orange line.

Figure 4.9 shows how at the end of the landing phase the joint position reaches the recalculated desired position when the stiffness control is applied, as opposed to the joint positions without stiffness control where the error remains high in steady-state.

Error comparison is depicted in Figure 4.10 where the blue line is the error position of joints with stiffness controller strategy, the orange line depicts the error joint position without any control action. The blue line shows how the error is driven to zero in steady-state, the Figure 4.10 illustrates how the stiffness control helps to reduce error position.

Finally, Figure 4.11 illustrates the position of the CoM through the landing phase. The principal aim of stiffness control system is to keep CoM close to the zero point within the contact base.

This figure shows how the CoM is displaced from the zero position in the landing stage. However, the stiffness control system drives the CoM close to the zero point. Also, it reduces oscillations and large displacements from the zero point. In Figure 4.11, the blue line depicts the CoM with stiffness control and the orange line is the CoM without control actions.

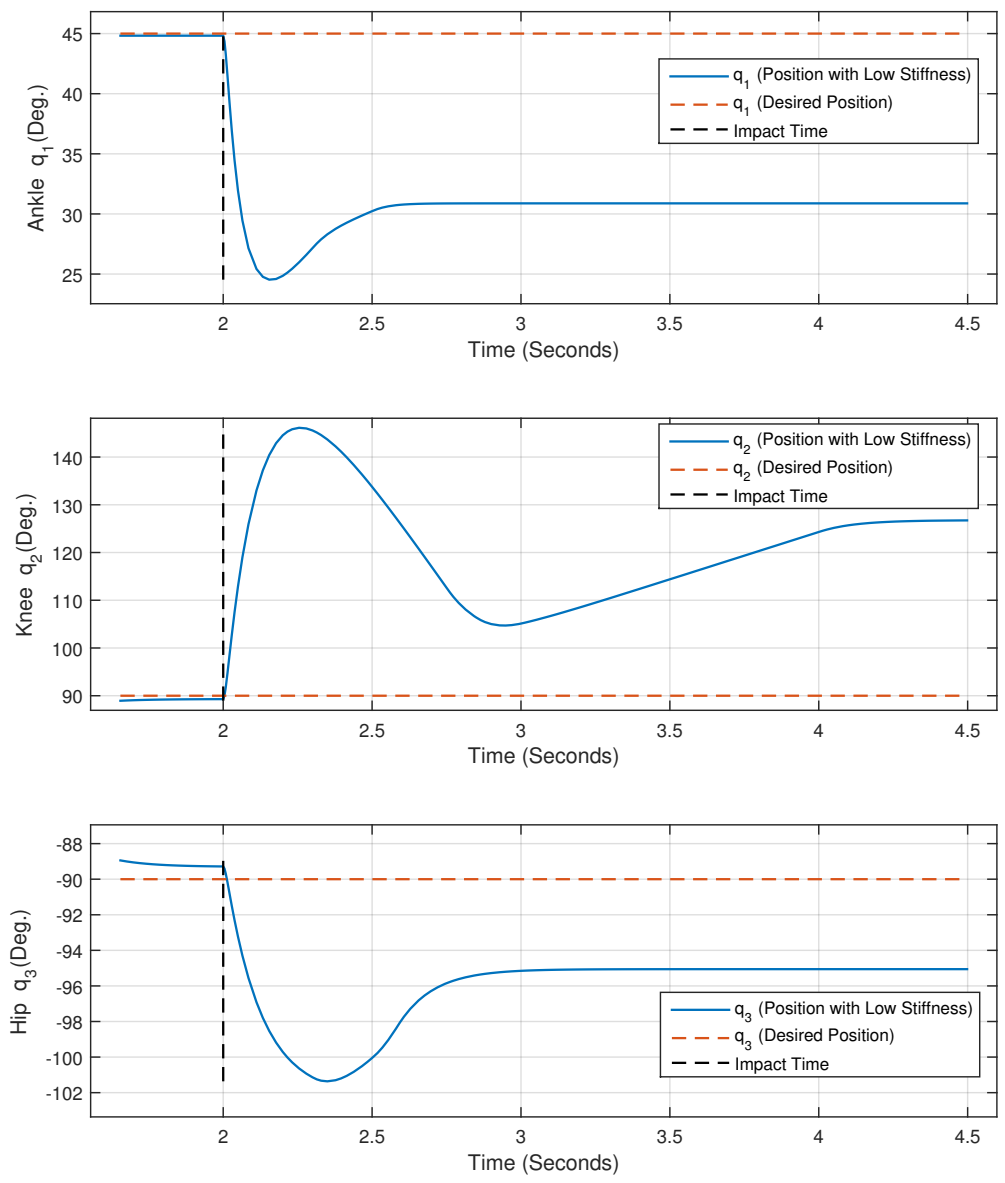


Figure 4.7: Joint position with low stiffness

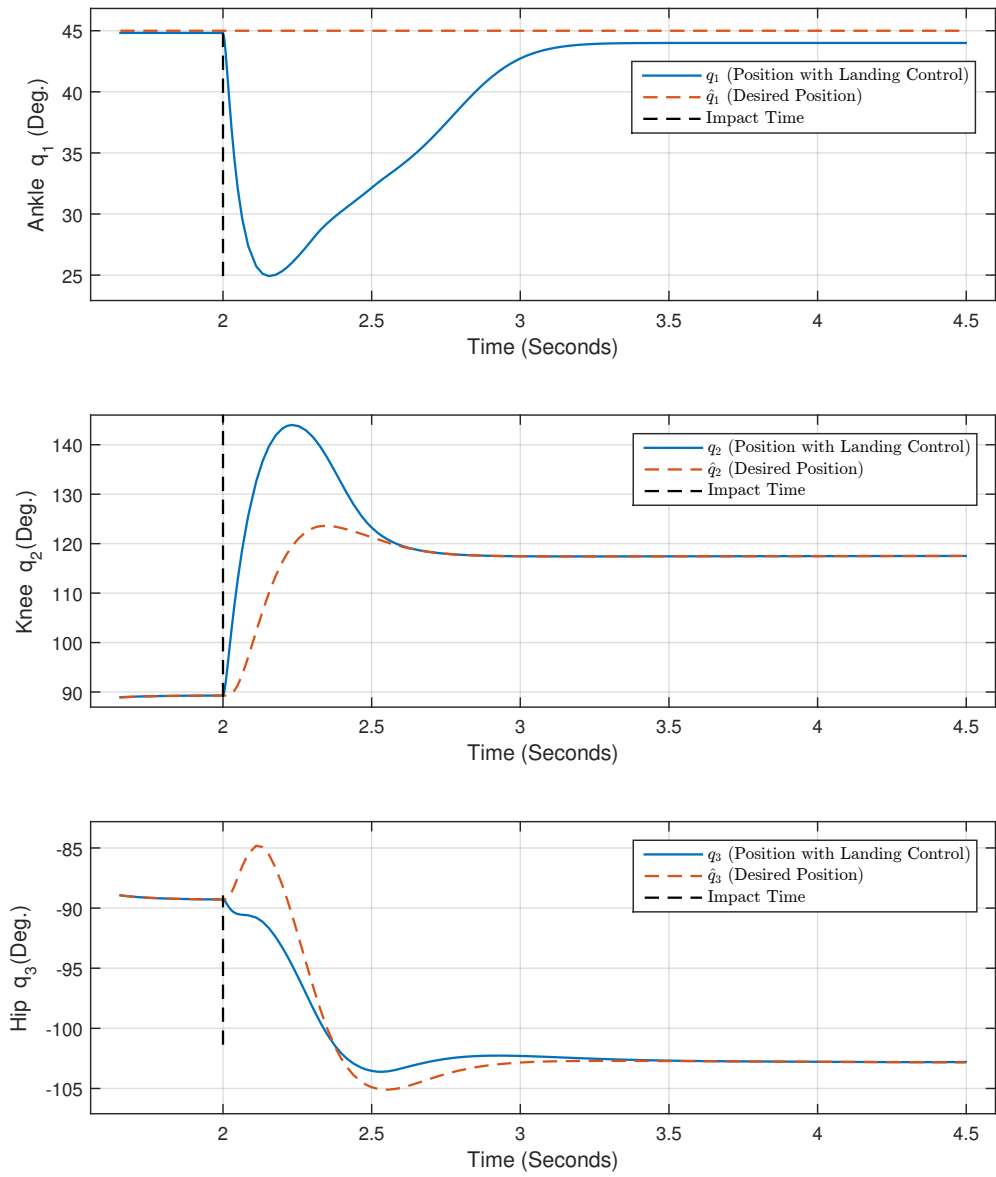


Figure 4.8: Joint positions with control landing

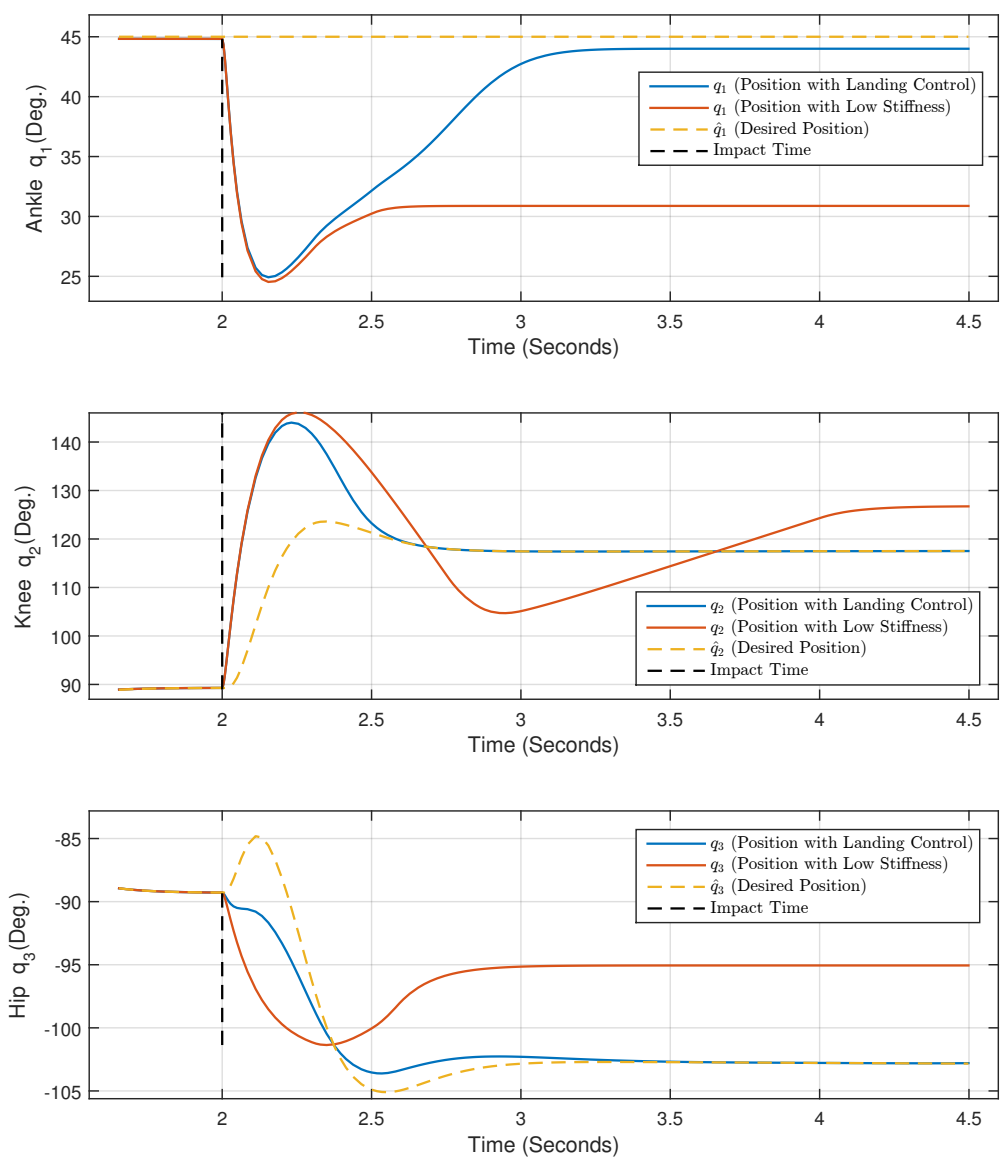


Figure 4.9: Comparison of joint position with and without control

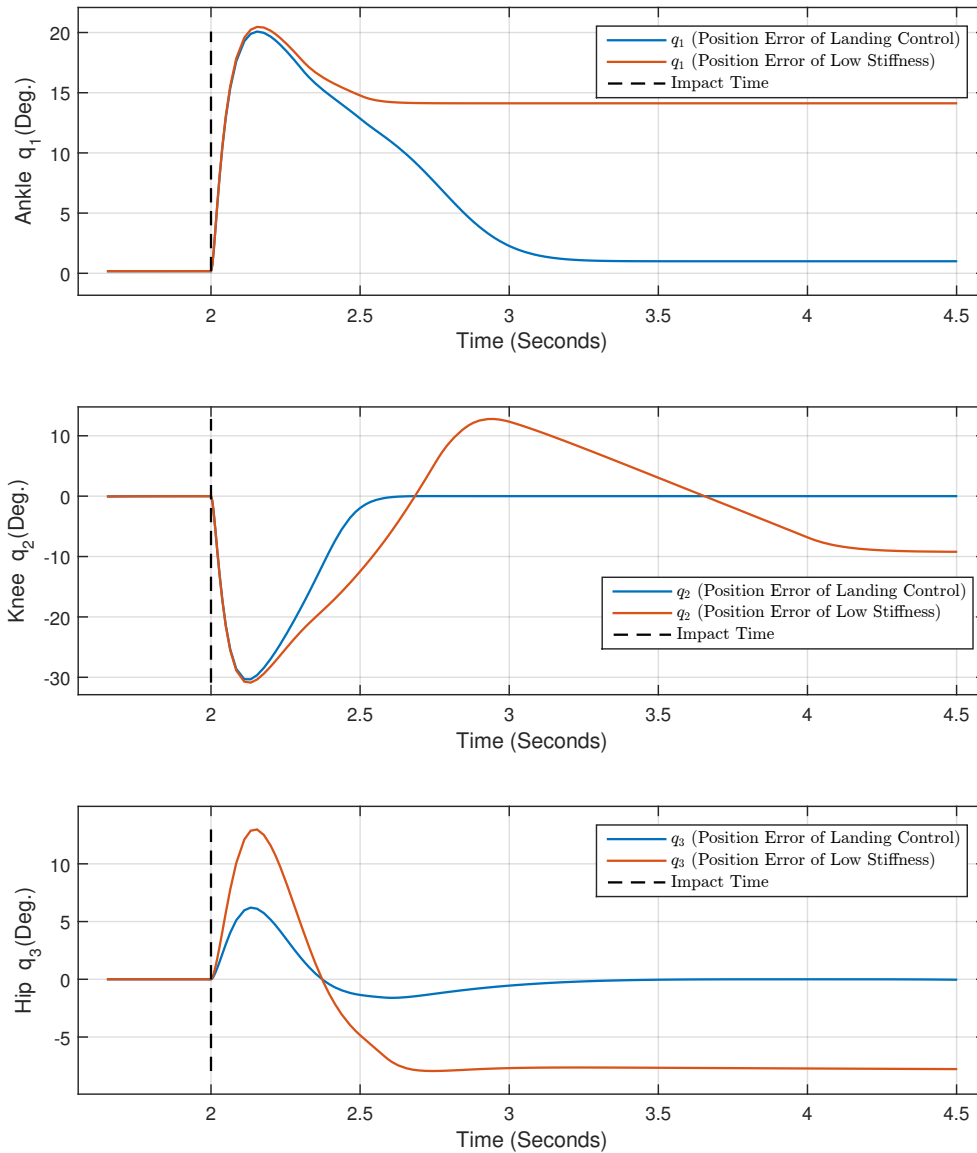


Figure 4.10: Position error for joint with low stiffness with and without control

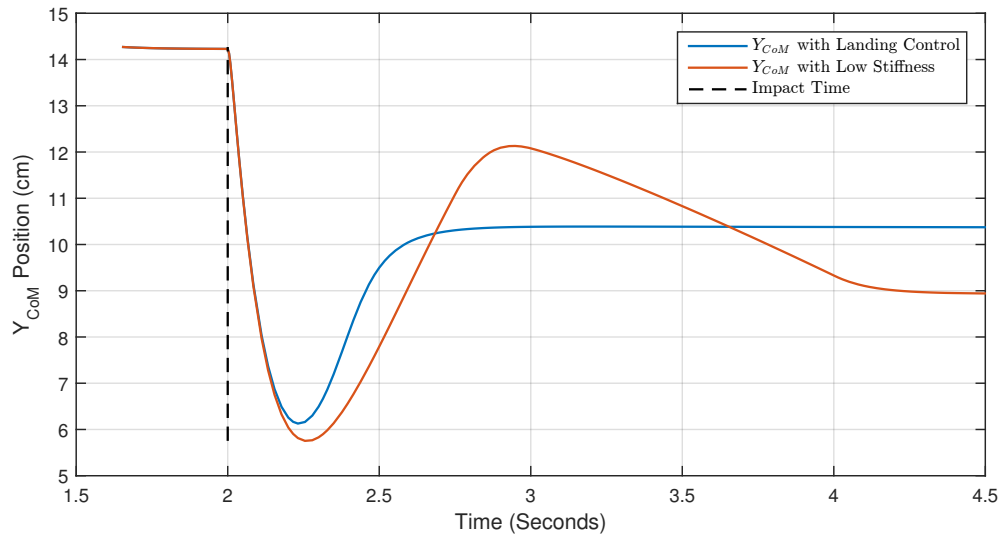
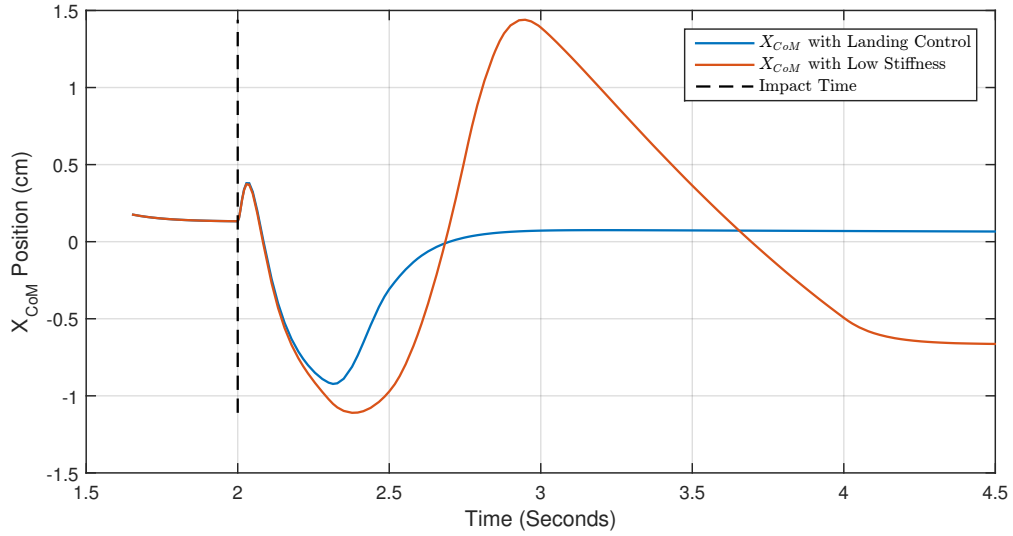


Figure 4.11: Comparison of the CoM position for robot with and without control

4.5 Falling Forward Motion

As mentioned in the introduction of this dissertation, during the last three decades the research and development of humanoid robots have increased exponentially. One of the most outstanding advances has been the use of the linear pendulum model for balance walking control. This advance has had excellent outcomes, however sometimes balance control can fail and the robot falls. Therefore, it is necessary to answer a new question: what must the robot do when falling is inevitable? Some authors describe different kinds of a possible solution to this problem. The consensus is that the main goal is to prevent or minimize damage to the robot at the moment of hitting the ground.

Taking into account a bio-inspired approach, when humans are falling, they try to protect their chest and head using their hands and arms. Consequently, the most common fractures in human fall accidents are the wrist, arm, or clavicle. According to Ruiz-del Solar [54], it is very common for humans to break their bones due to falling, however for humanoid robots, the links are particularly strong and the impact force affects the joints which in turn damage the motors. Using this last idea, once the robot detects that it is falling, it moves its arms to the front position trying to protect the head and chest from the impact force. However, this movement puts the integrity of the arm motors at risk. To protect the motors, the stiffness of every arm motor is decreased, affecting the gains of the motor control system. The PID control gains are then calculated using the Optimal Low Stiffness approach and these values are programmed into the PID controller of the robot motors. The falling impact reduction strategy is based on reducing the motor stiffness immediately before the robot impacts the ground. In the next section the falling strategy is explained.

4.5.1 Fall Strategy

The aim of the fall algorithm is to reduce the damage in the humanoid robot due to falling. Once the fall is detected, the robot moves its arms to the front position. The stiffness of the arm motors is then reduced trying to absorb the impact force in order to protect the robot and the arms. Rather than relying on a complex kinematic model and an accurate estimate of the robot global position to plan a trajectory that reduces the impact forces, the proposed algorithm is based on reducing the stiffness to trade off position displacement for damping capabilities.

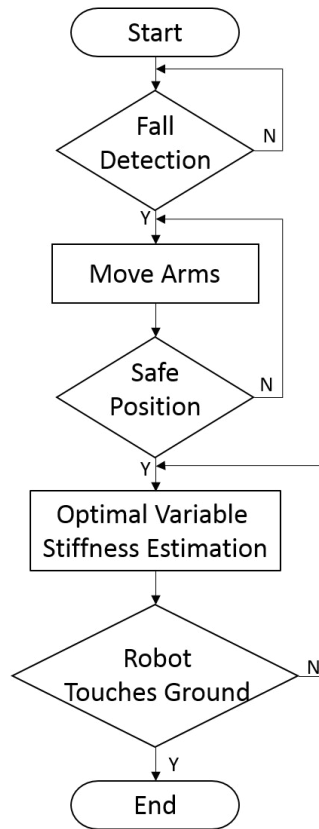


Figure 4.12: Flow diagram of fall algorithm

4.5.1.1 Fall Detection

The robot has an Inertial Measurement Unit (IMU), which is normally composed of an accelerometer and a gyroscope. With the IMU it is possible to measure the inclination angle and acceleration of the body of the robot. The limit falling angle is the minimum angle that takes the projection of the CoM out of the foot support area. If the inclination angle is larger than the minimum angle it is assumed that the robot is starting to fall, see Figure 4.13. The limit angle of inclination can be calculated experimentally or by using the projection of the CoM.

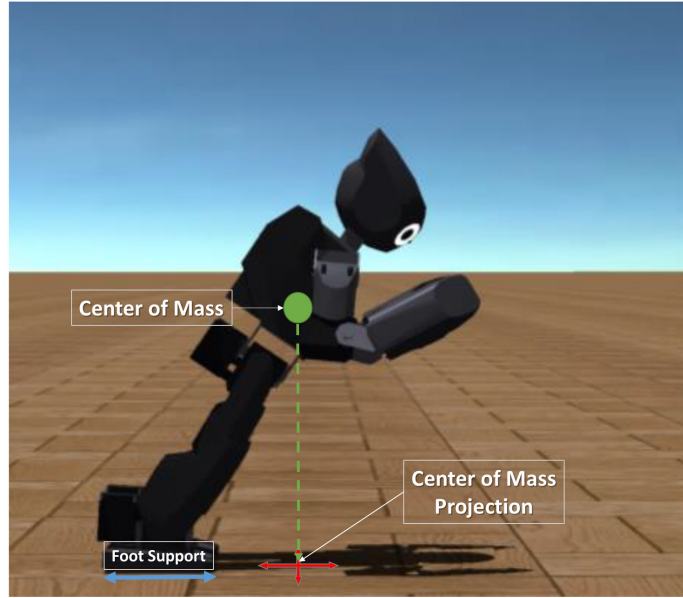


Figure 4.13: CoM projection

4.5.1.2 Arms Position

Once it is determined that the robot is falling, the robot arms move to the safe position, that is, to impact the ground first, protecting the more expensive components, such as the main processing board or sensors located in the head and torso.

4.5.1.3 Optimal Variable Stiffness Approach

Once the safe position is reached, new PID gains are calculated using the Optimal Variable Stiffness approach presented in this dissertation in section 4.4.1.4. These new PID gain values reduce the stiffness of the elbow and shoulder motors. The small stiffness allows the absorption of the impact force by the arms.

CHAPTER 5

OPTIMAL VARIABLE STIFFNESS PERFORMANCE EVALUATION

5.1 Note to the Reader

Part of this chapter was published in the IEEE-SoutheastCon [26] and the proceedings of the following conference: Innovations in Bio-Inspired Computing and Applications 2015 [27].

5.2 One-legged Jumper Robot Platform

The One-legged jumper robot platform is a three DoF robot as shown by Figure 5.1. The actuator of every joint is an MX-28 motor by Robotis Inc. The actuator's weight is 72g, and it can provide a maximum torque of 3.1 Nm. The motor runs in "endless turn" mode.

The motor provides feedback about the angular position with a 0.088° resolution, the velocity, and the joint torque. Additionally, the motor has a 32-bit Cortex-M3 microcontroller unit, where a programmable PID controller is implemented. The Controller is implemented using a CM-2 board with an Atmega128 CPU. This board communicates with the PC via the RS232 serial interface, and with the motors via the RS485 serial protocols. Table 5.1 shows the robot link parameters used in the kinematic and dynamic model.

Table 5.1: Parameters of One-legged Robot

	Length $L_i(m)$	Mass $m_i(Kg)$	Center of Mass $L_{ic}(m)$
Foot	0.010	0.051	0.045
Shank	0.098	0.051	0.072
Thing	0.098	0.051	0.072
Trunk	0.080	0.153	0.063

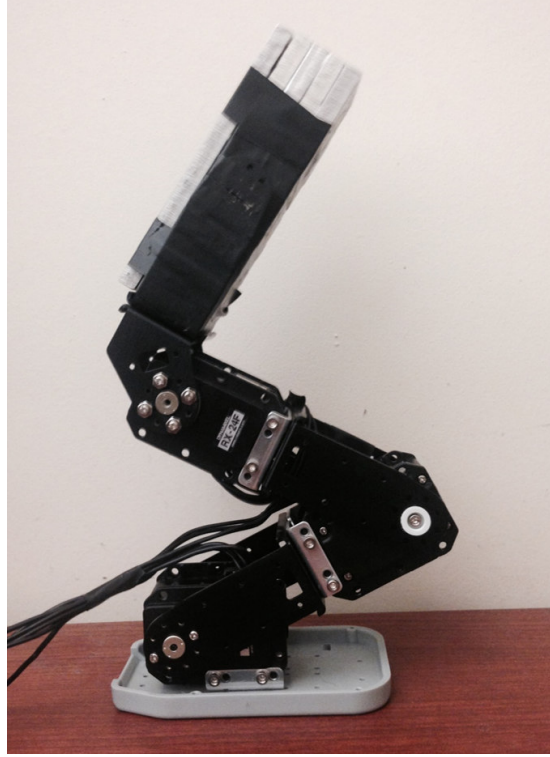


Figure 5.1: One-legged jumper robot platform

5.3 Experiments for Basic Jumping Motion in the Robotic Platform

In order to perform experiments to validate vertical jumping movements in the robotic platform, the Computed-Torque Control and Optimal Variable Stiffness were implemented in a CM-2 board and external PC. The experiments consist on the execution of two different movements. The first one consists of a vertical jump, and the second one consists of a free falling motion from different heights. Each experiment consists of several trials with different values for desired landing and impact velocity in the Fuzzy logic approach and with different values for ρ for the optimal control approach. Figure 5.2 shows the robot performing the vertical jump.

5.3.1 Take-Off Phase

Another important aspect to analyze is the CoM displacement during the takeoff and landing phases. The CoM displacement on the take-off phase is one of the jump requirements as explained in section 4.3.1. The idea is to keep the projection of the CoM close to the point zero within the contact surface. The zero

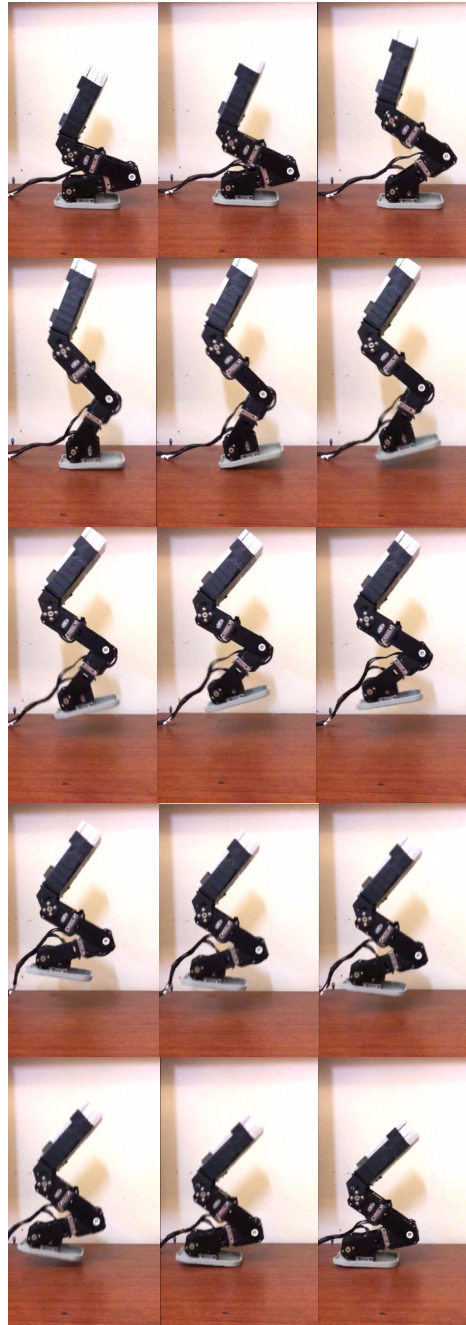


Figure 5.2: Vertical jump performed in the first experiment

point of the contact surface is basically the central point of the robot foot. The zero point is shown in Figure 5.3.

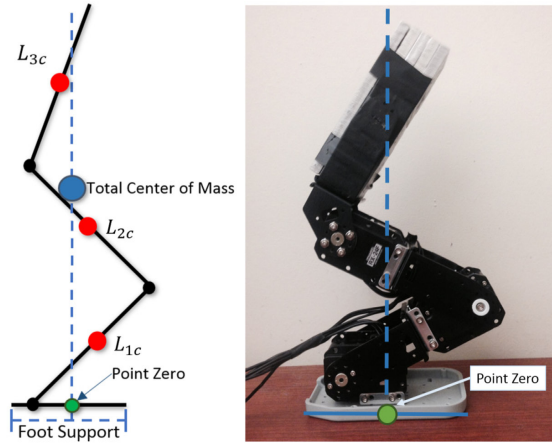


Figure 5.3: Point zero of foot support

The CoM displacement in the take-off phase indicates that there is an error in the DC motor trajectory. The trajectory error can be produced by two factors. The first factor could be due to errors in the calculation of the desired trajectory (section 4.3.1). The second factor is the error in the trajectory tracking of the DC motor. This can occur because some DC motor does not have the capability to track this kind of trajectories. The trajectory tracking sometimes is affected by some acceleration limitations of the DC motor.

Figure 5.4 shows the deviation of the CoM from the zero point in the take-off phase. Several trials were performed with the average deviation of the CoM values equal to 0.38 cm. This means that the Computed-Torque Control has a good performance following the original trajectory.

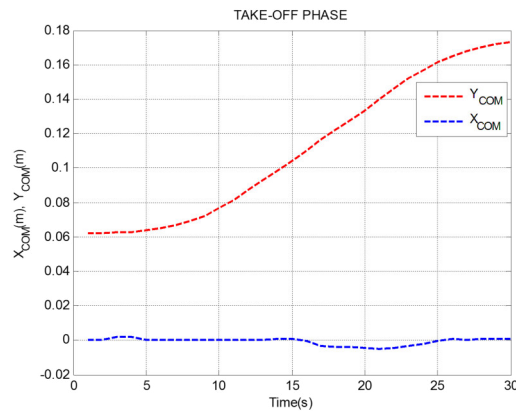


Figure 5.4: CoM in take-off phase

5.3.2 Landing Phase

The goal of this section of the basic jump experiment is to test the impact force reduction through the variation of motor stiffness in the landing phase. The impact force is estimated using Eq. (5.1), where d is assumed to be the difference between the touchdown position and the lowest position reached by the CoM during the landing phase. V_l is the impact velocity or the velocity at the moment of the impact and m is the robot mass.

$$F_{i-avg} = \frac{\frac{1}{2}mV_l^2}{d} \quad (5.1)$$

Large values of the displacement (d) in the y axis of CoM in the landing phase correspond to the soft landing and a low impact force, whereas small values of d correspond to a hard landing and a high impact force. Figure 5.5 shows the comparison between the CoM positions of two different jumps attempts. The first one uses a low stiffness, blue line, resulting in a significant displacement of the CoM from the touchdown position. The second one uses full stiffness, red line, and shows how the CoM stays near the touchdown position throughout the entire landing phase.

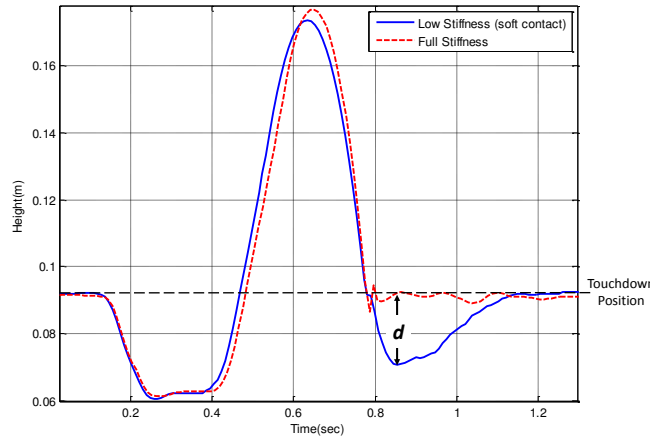


Figure 5.5: CoM position in a vertical jump using low and full stiffness

The landing velocity is calculated using Eqs. (5.2) and (5.3), since both the takeoff velocity and the maximum height are known previously to the motion.

$$y_{com}(t) = -\frac{1}{2}gt^2 + V_{to}t + Y_{to} \quad (5.2)$$

$$y_{max} = -\frac{V_{to}^2}{2g} \quad (5.3)$$

5.3.2.1 Landing Phase with Different Values of ρ

The vertical basic jump experiment described previously were implemented using the Optimal Variable Stiffness approach. The experiment was executed using different values of ρ while trying to keep the impact velocity constant. The impact velocity remains largely similar, thanks to fact that the Computed-Torque Control is performing the jump with the same set of parameters in every attempt. The CoM position for different levels of ρ is shown in Figure 5.6.

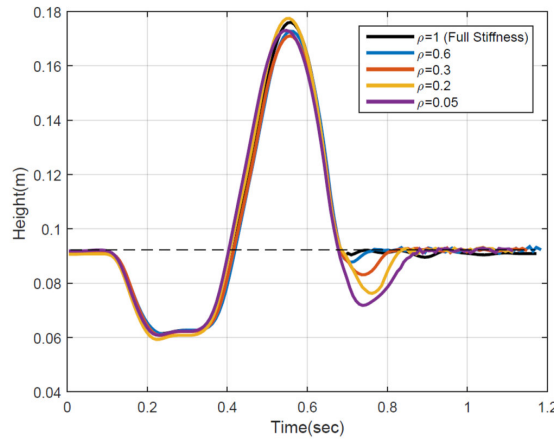


Figure 5.6: CoM position in vertical jump with different values of ρ .

Figure 5.6 depicts the CoM position during the whole vertical jump motion. Every phase of the vertical jump is clearly recognized, especially the landing phase. Figure 5.6 shows several trials with different ρ values. According to Figure 5.6, lower values of ρ increase the distance d , which means the impact force average was reduced. The results obtained by the average impact force, impact velocity, maximum height reached and desired landing are shown in Table 5.2. Table 5.2 shows how the impact force is reduced according to the ρ level, while the impact velocity is held almost constant. Additionally, results show that the Computed-Torque Control is allowing experiment repeatability, as the values of landing velocity and maximum height have a low variance between trials.

Table 5.2: Parameters of Figure 5.6

ρ	Max. High (<i>cm</i>)	Impact Velocity (<i>m/s</i>)	<i>d</i> (<i>cm</i>)	Impact Force Average (<i>Kg * m/s²</i>)
1	2.61	1.280828	N.A.	N.A.
0.6	2.21	1.256105	0.46	52.7877
0.3	2.23	1.241982	0.921	25.7756
0.2	2.65	1.288456	1.607	15.8987
0.05	2.25	1.257665	2.048	11.8861

5.4 Free Fall Experiment

The second experiment consists of a free fall movement. The objective of this experiment is to test the impact force reduction using different values of impact velocity and ρ parameter. The main reason to implement a free fall was to test the proposed method with greater impact velocities, as the impact velocity is limited by the maximum height reached by the robot in the vertical jump.

5.4.1 Free Fall Experiment with Different ρ Values

This experiment was performed with the aim of testing the effect of different impact velocities on the Optimal Variable Stiffness method. Figure 5.7 shows the CoM position in a free fall from 25 cm of height with different values of ρ .

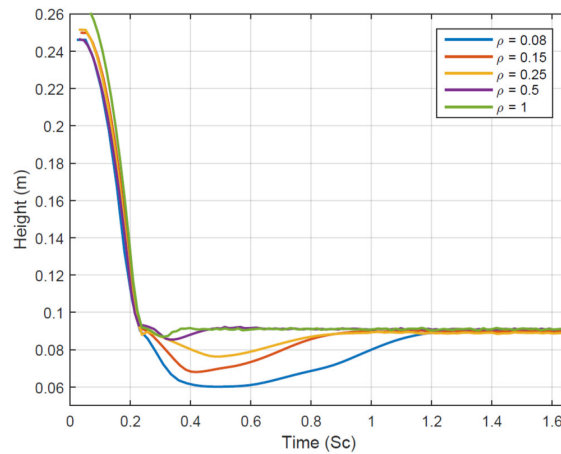


Figure 5.7: CoM position in free fall with different values of ρ .

Figure 5.7 shows a similar system behavior to the one shown by the first experiment of the vertical jump. The free fall height starts from 25 cm and the reached impact velocity is 1.78 m/s. Figure 5.7 shows how the distance d increases with small values of ρ , which means the impact force average was decreased.

5.4.2 Free Fall Experiment with Different Height Values

The second part of the experiment is shown in Figures 5.8 and 5.9, where the free fall is performed using different heights for the same ρ value. Figure 5.8 shows free fall motions from 20cm, 25cm, and 30cm of height using a ρ value of 0.15 and Fig. 5.9 shows the same experiment using a ρ value of 0.5.

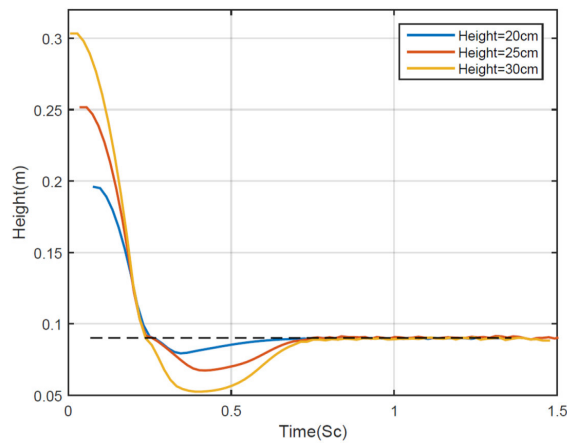


Figure 5.8: CoM position in free fall movement with $\rho = 0.15$.

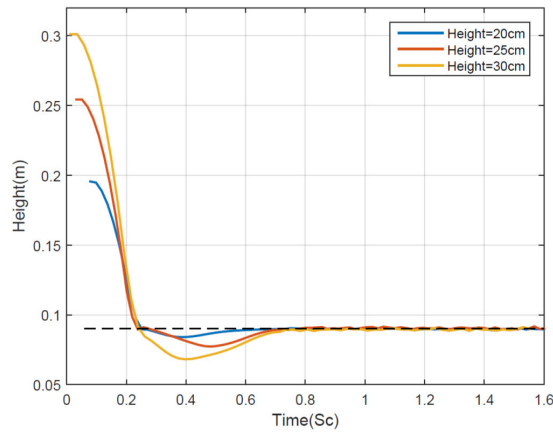


Figure 5.9: CoM position in free fall movement with $\rho = 0.5$.

Figures 5.8 and 5.9 show how the distance d increases when the height and impact velocity are large. The results of the second experiment using different values of ρ and heights are shown in Table 5.3.

Table 5.3: Parameters of Free Fall Experiment with Optimal Approach

ρ	Height=20cm		Height=25cm		Height=30cm	
	Imp. Vel.=15.4m/s	Imp. Force	Imp. Vel.=17.8m/s	Imp. Force	Imp. Vel.=21.32m/s	Imp. Force
-	d	Imp. Force	d	Imp. Force	d	Imp. Force
0.08	2.35	15.34	2.80	17.19	3.31	20.93
0.15	1.04	34.66	1.10	43.78	1.30	53.14
0.3	0.81	44.50	1.00	48.15	1.25	55.27
0.5	0.61	59.09	0.70	68.79	0.93	74.29

5.5 Fall Forward Experiment with Humanoid Robot

A Darwin-Op Humanoid Robot was chosen to perform experiments to test the falling strategy. This robot is one of the most popular humanoid robots used by many research labs around the world. Additionally, the Darwin-OP uses Dynamixel MX-28 motor actuators. This motor actuator has some special features such as absolute encoder resolution of 4,096, stall torque of 31.6 Kg-cm and a programmable PID control. The last capability is the most important because the present work proposes the design of an optimal PID gain controller in order to generate variable stiffness.

The experiment consists of running several trials of the falling robot with different values of ρ . The robot is standing up and it is pushed from the back to fall forward. The control strategy will then command the movement of the arms to the front safe position. The new PID gain values according to ρ are programmed for the elbow and shoulder motors. Figure 5.10 shows a motion sequence of the experiment.

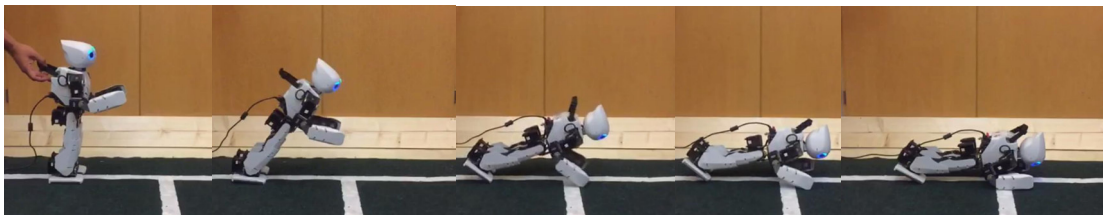


Figure 5.10: Motion sequence of the experiment

At the end of the motion sequence shown in Figure 5.10, the arms are displaced from the front safe position in order to reduce the impact force between the robot body and the ground. Figure 5.11 shows the

position of the CoM of the robot in five different trials. Every trial has different values of ρ , $\rho = 0.05, 0.1, 0.3, 1, \text{ and } 3$.

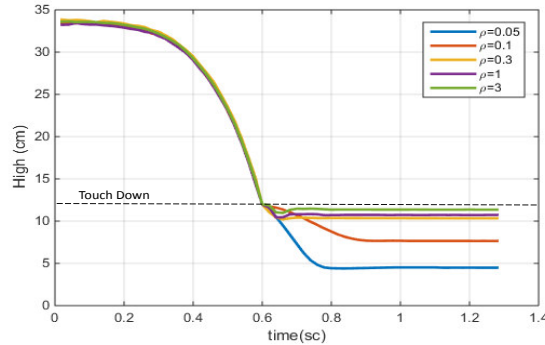


Figure 5.11: CoM position with different ρ values

Another important parameter used to measure the ground impact force is the displacement (d) of the CoM at the moment of impact. Figure 5.11 shows how d increases as ρ is reduced. The average impact force can be estimated using Eq. (3.5), where d is inversely proportional to the impact force.

The results of the impact force, impact velocity, and CoM displacement are depicted in Table 5.4.

Table 5.4: Experiment Results

ρ	Impact Velocity (m/s)	Distance (d) (cm)	Average Impact Force (N)	Impact Reduction (%)
0.05	1.78	7.61	60.44	86.8
0.1	1.72	4.34	98.84	78.52
0.3	1.75	1.79	248.07	41.60
1	1.72	1.54	278.55	39.46
3	1.69	0.9	460.14	0

Table 5.4 depicts the results of the five trials shown in Figure 5.10. Results show how the displacement (d) increases according to the decrements of ρ . The impact force is reduced with smaller values of ρ and it increases with large values of ρ . This highlights the importance of the stiffness variation to reduce the impact force. The low stiffness generates large values of displacement and a significant reduction of the impact force.

CHAPTER 6

FUZZY VARIABLE STIFFNESS APPROACH

6.1 Note to the Reader

Part of this chapter was published in the IEEE-SoutheastCon [26] and the proceedings of the following conference: Innovations in Bio-Inspired Computing and Applications 2015 [27].

This chapter will depict the second approach to generate variable stiffness based on Fuzzy logic theory .

6.2 Variable Stiffness with Fuzzy Logic Approach

The Fuzzy logic strategy uses two input parameters based on the impact velocity and desired impact force variables. The output of the Fuzzy logic system is the value of the proportional gain to be used in the control loop of the DC motor.

The variable stiffness capability is produced through the variation of the K_P gain of the PID controller. Small K_P gain value implies low stiffness in the DC motors. The derivative constant gain value, K_D , of the PID controller is calculated for a critical damping response where $D = 2\sqrt{P}$.

The impact force is proportional to the squared velocity reached by the robot at contact, in accordance with Eq. (3.5). Based on the K_P gain effect on the stiffness and the impact force, a Fuzzy system is proposed in order to estimate the adequate K_P value to implement a soft fall contact capability.

The proposed control approach is a Mamdani Fuzzy inference system [63], composed by two inputs and one output using min and max T-norm operators as shown in Figure 6.1. The inputs of the Fuzzy logic strategy are Desired-Impact force and Impact-Velocity. The output is the stiffness level which is proportional to the K_P gain value. The Fuzzy logic strategy is explained in terms of the fuzzifier, the set of rules and the defuzzifier.

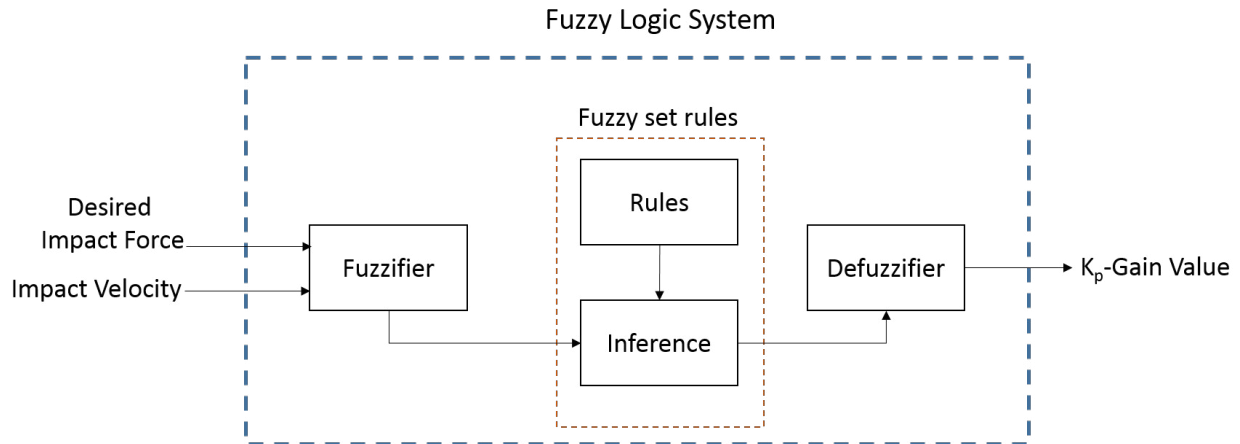


Figure 6.1: Fuzzy logic system

6.2.1 The Fuzzifier

The fuzzifier has two inputs. The first one is the Desired Impact Force input which is related to the impact force exerted by the ground on the robot at landing. Desired impact force values close to 1 imply a big impact forces, and values close to 0 are associated to small impact forces. Desired impact force is not a measurable input variable; however it is a preset value according to the desired reduction of the impact force at the end of the movement.

The impact force is calculated using Eq. (3.5), and it is used to evaluate the system performance. Thus, when a low value of "Desired impact force is set up, a low level of Impact Force is expected. The universe of discourse of this variable has values between 0 and 1. This input is divided in 5 triangular member function distributed symmetrically through the universe of discourse as depicted in Figure 6.2. The membership functions are Soft, Very Soft, Medium, Hard and Very Hard.

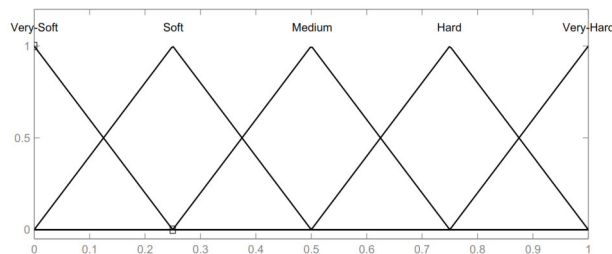


Figure 6.2: Desired-impact force

The second input is called Impact Velocity. This variable is calculated by Eqs. (3.2) and (3.3) immediately after the take-off phase using the value of the take-off velocity when the movement is a vertical jump. When the movement is a free fall the impact velocity can be calculated using the height where the robot is starting to fall. Additionally, an Inertial Measurement Unit (IMU) can be used to measure the velocity for both types of movements. This input has five triangular membership functions called Very Low, Low, Medium, High, and Very high. They are spread uniformly between 0 and 3m/s as depicted in Figure 6.3. The maximum limit of the Impact Velocity input is 3m/s and it corresponds to the velocity reached by the robot in free fall from 0.5 m of height.

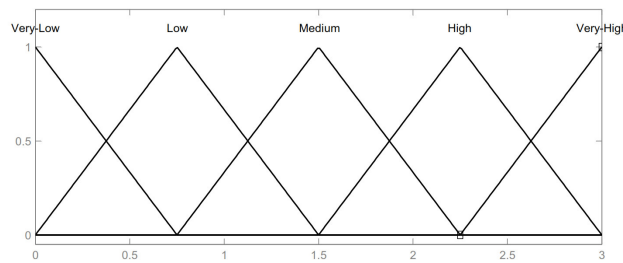


Figure 6.3: Impact-velocity input

Finally, the output Stiffness estimates the K_P gain value that will be applied to the ankle and knee motors. It is divided into five triangular membership functions uniformly distributed called: Very Low, Low, Medium, High, and Full. This output has values between 0 and 32 as shown in Figure 6.4. These limits correspond to the minimum and maximum K_P gain value applied to the motor without oscillations (overshot).

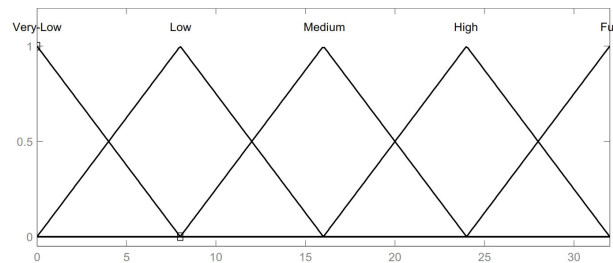


Figure 6.4: Stiffness output (K_P gain-value)

6.2.2 Fuzzy Set of Rules

The Fuzzy set of rules determines the relation between output to the inputs. In this case, these rules take the input set and make calculations to find the adequate output set. Every rule has the form depicted by Eq. (6.1). The rules calculate the min-product between the Fuzzy value of two inputs, the Desired impact force and the Impact Velocity, and it projects this value over the assigned Fuzzy output set. The outcome of the rule is the area of the output set below of the min value of the inputs, i.e. min-product between output and min value of inputs. The final outcome is the max-product among all outputs of the complete set of rules. The complete set of rules is composed of 25 rules. At the end, the general outcome of the Fuzzy set rules is an area composed by the max-products of the partial output of every rule of the complete set.

$$\mathbf{if} (Input(1) \text{ is } mf(i)) \mathbf{and} (Input(2) \text{ is } mf(i)) \mathbf{then} Output \text{ is } mf(i) \quad (6.1)$$

The complete set of rules is described in Table 6.1.

Table 6.1: Set of Rules

		Impact-Velocity				
		Very-Low	Low	Medium	High	Very-High
Desired Imp-Force	Very-Soft	Low	Low	Very-Low	Very-Low	Very-Low
	Soft	Medium	Medium	Low	Low	Low
	Medium	Hard	Hard	Medium	Medium	Medium
	Hard	Full	Hard	Hard	Hard	Hard
	Very-Hard	Full	Full	Full	Hard	Hard

6.2.3 Defuzzifier

The defuzzifier is used to calculate the final value of a general output of the Fuzzy logic system. The input for the defuzzifier is the area calculated by the Fuzzy set rules. The aim of the defuzzifier is to calculate final value of the output. For the current case, the output is the proportional gain of the control loop.

The centroid method is used by the defuzzifier to estimate the final value of the K_P gain value. The general formulation of the centroid method is depicted by Eq. (6.2).

$$K_P^* = \frac{\int \mu_B(k_p) dk_p}{\int \mu_B dk_p} \quad (6.2)$$

where K_P^* is the centroid value representing the proportional gain of the PID controller; μ_B is the membership value, in this case the value of the area in the point k_p ; k_p is one point into the universe of the variable K_P , see Figure 6.5.

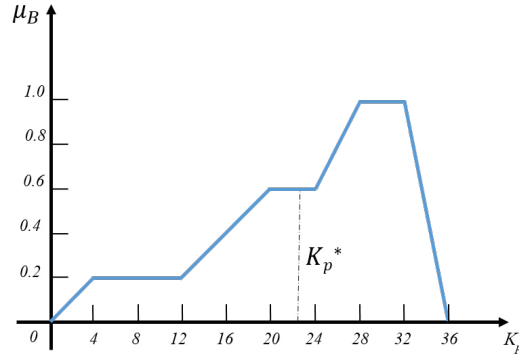


Figure 6.5: Centroid method

Finally, Figure 6.6 shows the Fuzzy surface. This surface shows the relation between inputs and the final output value. This is the result of calculation for every possible input value combination. The surface shows how all possible output values are into the range of usual proportional gain, values between 0 and 32 for the One-legged robot motor.

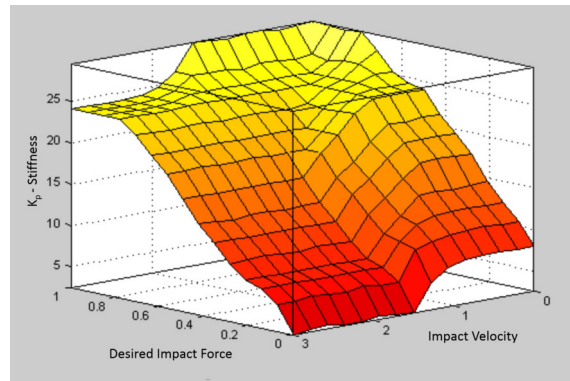


Figure 6.6: P-Gain value estimation surface

6.3 Experiments for Vertical Jump and Free Fall Motions Using Fuzzy Logic Approach

This section shows experiments focused on the landing phase of the jumping motion using just the Fuzzy approach. The Fuzzy approach has the same objective of the optimal approach, it is focused on the generation of variable stiffness to reduce the impact force. According to the last statement, the take-off phase and fall forward experiments are not studied in this chapter.

Vertical jump and free fall experiments described in chapter 5 were implemented using Fuzzy variable stiffness approach. For every attempt, the inputs to the Fuzzy variable stiffness system were established according to the desired impact force and impact velocity estimated as previously explained.

6.3.1 Experiments for Vertical Jump with Constant Impact Velocity and Variable Desired Impact Force

The experiment was executed using different values for the desired impact force while trying to keep the impact velocity constant. The impact velocity is kept fairly fixed thanks to the Computed-Torque Control applied in the take-off phase, similar to that explained in Chapter 5. The CoM position for the different Desired Impact Force values is shown in Figure 6.7.

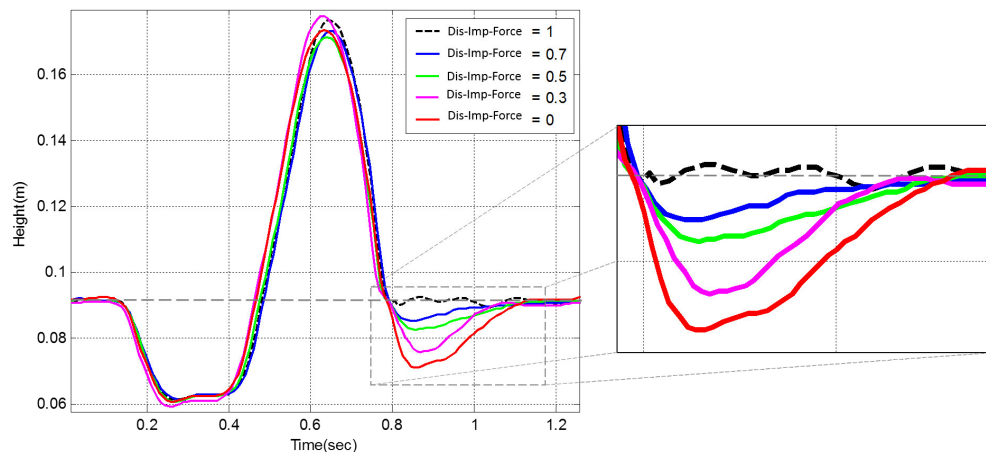


Figure 6.7: CoM position in vertical jump with different values of desired impact force input.

Figure 6.7 shows several trials with different desired impact force values. According to Figure 6.7, lower values of desired impact force increase the distance d , which means the impact force average was reduced.

Table 6.2: Parameters of Figure 6.7

Desired Imp-Force	Max. High (cm)	Impact Velocity (m/s)	d (cm)	Impact Force Average ($Kg * m/s^2$)
1	2.66	1.298307	N.A.	N.A.
0.7	2.34	1.273923	0.6	41.38344
0.5	2.16	1.260041	0.855	28.40968
0.3	2.79	1.308083	1.528	17.13320
0	2.35	1.274692	2.009	12.37434

The results are very similar to those obtained with the Optimal Variable Stiffness approach, previously shown.

The obtained average impact force, impact velocity, and maximum height reached are shown in Table 6.2. Table 6.2 shows how the impact force is reduced according to the desired impact level, while the impact velocity is held almost constant.

6.3.2 Experiment for Free Fall Motion Using Fuzzy Variable Stiffness

The second experiment is a free fall motion. This experiment is performed with the objective to test the impact force reduction effect according to variations of the Impact Velocity and Desired impact input variables.

6.3.2.1 Experiment for Free Fall Motion with Variation of Desired Impact Force Input

It was performed with the aim to test the effect of different impact velocities on the Fuzzy Variable Stiffness method. Figure 6.8 shows the CoM position in a free fall from 25 cm of height with different Desired Impact Force input values.

Figure 6.8 shows similar experiments to those on section 5.4, where the robot falls from 25 cm with an impact velocity of 1.78 m/s. This figure shows how the distance d increases with small values of the Desired Impact Force, this means that the impact force average is decreased.

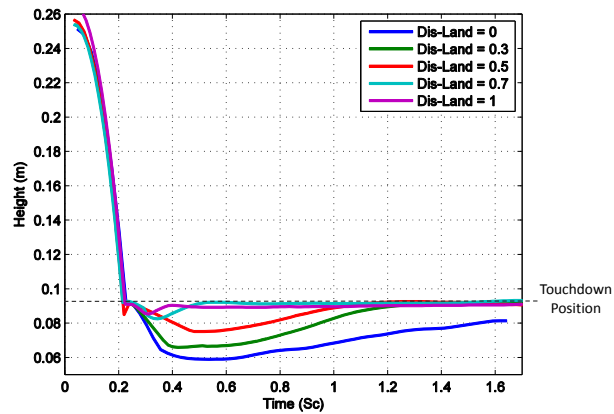


Figure 6.8: CoM position in a free fall with different values of desired impact force.

6.3.2.2 Experiment for Free Fall Motion with Variation of Impact Velocity

The second part of the free fall experiment tests the effect that different Impact Velocity values have over the impact force reduction. The results are shown in Figures 6.9 and 6.10, where the free fall is performed using different heights and the same Desired Impact Force input value. Figure 6.9 shows free fall motions from 20cm, 25cm, and 30cm of height using a Desired Impact Force value of 0.3. Figure 6.10 shows the same experiment using a Desired Impact Force Value of 0.7.

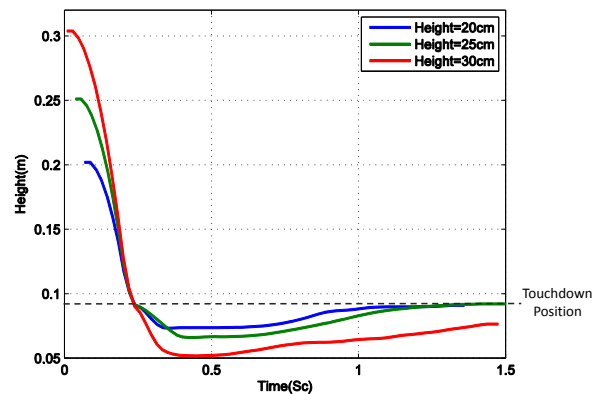


Figure 6.9: CoM position in free fall movement with desired impact force = 0.3.

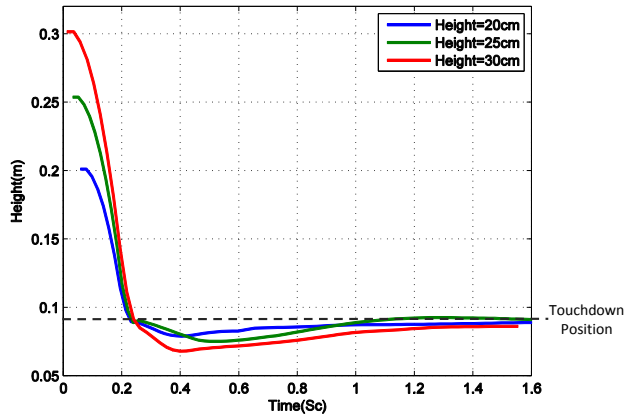


Figure 6.10: CoM position in free fall movement with desired impact force = 0.7.

Figures 6.9 and 6.10 show how the distance d increases when the height and impact velocity are large. The results of the second experiment using different values of Desired Impact Force and Impact Velocity are shown in Table 6.3.

Table 6.3: Parameters of Free Fall Experiment with Fuzzy Approach

Desired Imp-Force	Height=20cm		Height=25cm		Height=30cm	
	Imp. Vel.=15.4m/s	Imp. Force	Imp. Vel.=17.8m/s	Imp. Force	Imp. Vel.=21.32m/s	Imp. Force
-	d	Imp. Force	d	Imp. Force	d	Imp. Force
0.3	2.6	13.95	3.40	14.25	4.9	14.19
0.5	2.0	18.14	2.70	17.95	4.3	16.17
0.7	1.0	36.28	1.68	28.50	2	34.7
0.9	0.5	72.5	0.70	71.21	0.96	74.1

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

This chapter summarizes the most relevant findings and results of this dissertation and includes a number of ideas for future research in low stiffness generation.

7.1 Summary of Results and Findings

The current research proposes a framework to perform jumping motions with legged robots and reduce the impact force during the landing phase. The different stages of the jumping process were studied. In order to be able to perform the movement, the most important properties for each stage were analyzed.

- For the take-off phase, the acceleration of the robot CoM must be greater than gravity in order to detach from the ground. To ensure acceleration greater than gravity, the proposed system uses a trajectory generator based on the CoM and ZMP. A Computed-Torque Control approach is used to track the desired trajectories while reaching the required CoM acceleration. The use of the Computed-Torque DC motors Control approach and the trajectories based on ZMP and CoM conditions, allow the robot to jump. The jump motion was performed on a One-legged robot conformed by three DC motors located in the ankle, the knee, and the hip, without any mechanical devices, such as springs or dampers.
- The most important jumping stage is the landing phase, since it is the moment where it is mandatory to reduce the impact force to preserve the balance and the integrity of the robot. For the landing phase, two different approaches were proposed, both aiming to generate variable stiffness in the joint motors. The first approach is called Optimal Variable Stiffness and is based on an optimal control strategy. For this method, a ρ parameter was defined with the purpose of pre-multiplying the Q matrix and producing new PID gain values. With this method, low stiffness is reached with values of ρ

between zero and one. Additionally, this approach uses a control feedback loop to reduce the CoM displacement with respect to the zero point on the contact surface. This feedback keeps the CoM close to the zero point while improving the balance of the robot in the moment of landing. Several experiments were performed using a One-legged robot. The optimal method was tested performing several trials with different values for ρ . In all test cases, this method reduced the impact force according to the ρ parameter as shown in table 5.2.

The second approach used was labeled Fuzzy Variable Stiffness method. In this second method, the Fuzzy system estimates the K_P gain value of the PID controller for the landing phase allowing for displacement of the CoM at the touchdown position. The estimation of the K_P value was done using information from the impact velocity and the desired impact force. The proposed method based on Fuzzy logic theory allows for the reduction of the impact force when the robot lands on the ground. This method was tested performing several trials using different values for the desired impact force and the impact velocity. The desired impact force and the impact velocity are the two inputs of the Fuzzy logic system. The variation of these inputs can generate different levels of stiffness while generating different levels of impact force reduction as shown in table 6.2.

- Another important result obtained carrying out this research pertains to the robot balance during the landing phase. The robot balance is measured according to the displacement of the CoM with respect to the zero point of the foot support. For this performance metric, the Fuzzy approach presents a strong drawback, since this method does not include a feedback loop to reduce the displacement of the CoM with respect to the zero point of the contact zone. On the other hand, the optimal approach performed better while keeping the CoM close to the zero point. Although both methods can be used to reduce the impact force, the optimal approach has better balance performance than the Fuzzy method. However, the Fuzzy approach depends less on the motor model.
- Another result from this research is the application of the Optimal Variable Stiffness method to falling robots. In some cases, the balance control fails and it is inevitable for the robot to fall. It is then necessary to preserve the integrity of the robot and its components such as processors, motors, and sensors. An algorithm to protect the robot when it is falling was proposed. The robot integrity is

accomplished by applying the Optimal Variable Stiffness control method to reduce the impact force on the arm joints, shoulders and elbows as depicted in section 4.5.

7.2 Future Work

The future work is divided on two sections. The first section contains possible improvements to the implemented control strategies, and the second section depicts other applications for the Optimal Variable Stiffness and Fuzzy Variable Stiffness methods.

7.2.1 Control Strategies

The use of adaptive control theory to generate an automatic method to set up the α and ρ values in the Optimal Variable Stiffness method is expected to improve the applicability of the algorithm to more diverse platforms and external operating conditions.

In addition, the idea of incorporating three independent factors that modify the K_P , K_I and K_D gains of the PID controller. This would improve the systems sensitivity of the PID parameters.

The design and integration of a feedback loop control system to keep the robot balance during the Fuzzy Variable Stiffness approach, is expected to improve the performance of this method, without increasing its implementation complexity.

7.2.2 Applications

The Optimal Variable Stiffness approach could also be applied to reduce lateral disturbances and uneven terrain effects in the standing position of humanoid robots. Hence, the robot would be able to avoid oscillations created by the combination of disturbances and the rigidity of the robot body.

The Optimal Variable Stiffness could also be applied to the generation of compliance capabilities in the legs and arms prosthetic devices. This could make the use of the prosthetic devices seem more natural. Also these new capabilities would allow for the use of different stiffness values throughout a physical therapy. Furthermore, the stiffness values could be dynamically be adjusted within a session to perform personalized therapy.

Finally, the Optimal Variable Stiffness method could be used to generate soft contact in the walking motion of biped robots. Similar to the standing situation, soft contact would allow for the robot to cope with disturbances and uneven terrain. It would also protect the joints in the long term, by minimizing the peak impact force produced in each contact with the ground.

REFERENCES

- [1] G. A. Pratt, M. M. Williamson, P. Dillworth, J. Pratt, and A. Wright, “Stiffness isn’t everything,” in *experimental robotics IV*. Springer, 1997, pp. 253–262.
- [2] K. Pérez, E. Rojas, H. Báez, S. Rodríguez, J. López, C. Quintero, and J. M. Calderón, “Application of an educational strategy based on a soccer robotic platform.”
- [3] H. Kitano, S. Tadokoro, I. Noda, H. Matsubara, T. Takahashi, A. Shinjou, and S. Shimada, “Robocup rescue: Search and rescue in large-scale disasters as a domain for autonomous agents research,” in *Systems, Man, and Cybernetics, 1999. IEEE SMC’99 Conference Proceedings. 1999 IEEE International Conference on*, vol. 6. IEEE, 1999, pp. 739–743.
- [4] A. Weitzenfeld, J. Biswas, M. Akar, and K. Sukvichai, “Robocup small-size league: Past, present and future,” in *Robot Soccer World Cup*. Springer, 2014, pp. 611–623.
- [5] A. Eguchi, “Robocupjunior for promoting stem education, 21st century skills, and technological advancement through robotics competition,” *Robotics and Autonomous Systems*, vol. 75, pp. 692–699, 2016.
- [6] J. M. Calderón, E. R. Rojas, S. Rodríguez, H. R. Báez, and J. A. López, “A robot soccer team as a strategy to develop educational initiatives,” in *Latin American and Caribbean Conference for Engineering and Technology, Panama City, Panama*, 2012.
- [7] S. Rodríguez, E. Rojas, K. Pérez, J. Lopez, H. Baez, and J. M. Calderón, “Stox’s 2013 team description paper,” 2013.
- [8] J. Biswas, J. P. Mendoza, D. Zhu, S. Klee, and M. Veloso, “Cmdragons 2014 extended team description.”
- [9] S. Rodriguez, E. Rojas, K. Pérez, J. López, C. Quintero, J. M. Calderón, and O. Pena, “Stox’s 2015 extended team description paper,” *Joao Pessoa, Brazil*, 2014.
- [10] S.-J. Yi, S. McGill, H. Jeong, J. Huh, M. Missura, H. Yi, M. S. Ahn, S. Cho, K. Liu, D. Hong *et al.*, “Robocup 2015 humanoid adultsized league winner,” in *Robot Soccer World Cup*. Springer, 2015, pp. 132–143.
- [11] S. Rodriguez, K. Pérez, C. Quintero, J. López, E. Rojas, and J. Calderón, “Identification of multi-modal human-robot interaction using combined kernels,” in *Innovations in Bio-Inspired Computing and Applications*. Springer, 2016, pp. 263–273.
- [12] J. Ruiz-del Solar, “A dynamic and efficient active vision system for humanoid soccer robots,” *RoboCup 2015: Robot World Cup XIX*, vol. 9513, p. 316, 2016.

- [13] F. Martín, “Visual localization based on quadtrees.” ROBOT?2015 Second Iberian Robotics Conference.
- [14] A. K. Perez-Hernandez, A. Gomez-García, E. R. Rojas-Martínez, C. S. Rodríguez-Rojas, J. López-Jiménez, and J. M. Calderón-Chavez, “Edge detection algorithm based on fuzzy logic theory for a local vision system of robocup humanoid league,” *Tecno Lógicas*, no. 30, pp. 33–50, 2013.
- [15] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa, “Robocup: The robot world cup initiative,” in *Proceedings of the first international conference on Autonomous agents*. ACM, 1997, pp. 340–347.
- [16] C. Quintero, S. Rodríguez, K. Pérez, J. López, E. Rojas, and J. Calderón, “Learning soccer drills for the small size league of robocup,” in *Robot Soccer World Cup*. Springer, 2014, pp. 395–406.
- [17] J. Biswas, J. P. Mendoza, D. Zhu, B. Choi, S. Klee, and M. Veloso, “Opponent-driven planning and execution for pass, attack, and defense in a multi-robot soccer team,” in *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2014, pp. 493–500.
- [18] S. Rodríguez, E. Rojas, K. Pérez, J. López, C. Quintero, and J. Calderón, “Fast path planning algorithm for the robocup small size league,” in *Robot Soccer World Cup*. Springer, 2014, pp. 407–418.
- [19] J. P. Mendoza, J. Biswas, P. Cooksey, R. Wang, S. Klee, D. Zhu, and M. Veloso, “Selectively reactive coordination for a team of robot soccer champions,” *Proceedings of AAI-16 (2016, to appear)*, 2016.
- [20] D. H. Perico, I. J. Silva, C. O. V. Junior, T. P. Homem, R. C. Destro, F. Tonidandel, and R. A. Bianchi, “Newton: A high level control humanoid robot for the robocup soccer kidsize league,” in *Robotics*. Springer, 2015, pp. 53–73.
- [21] J. López, K. Pérez, E. Rojas, S. Rodríguez, J. M. Calderón, and A. Weitzenfeld, “Comparison between a fuzzy controller and classic controller applied to stabilize a humanoid robotic platform,” in *Advanced Robotics (ICAR), 2013 16th International Conference on*. IEEE, 2013, pp. 1–6.
- [22] E. Elibol, J. Calderon, M. Llofriu, C. Quintero, W. Moreno, and A. Weitzenfeld, “Power usage reduction of humanoid standing process using q-learning,” in *Robot Soccer World Cup*. Springer, 2015, pp. 251–263.
- [23] J. Song and J. Alves-Foss, “The darpa cyber grand challenge: A competitor’s perspective, part 2,” *IEEE Security & Privacy*, vol. 14, no. 1, pp. 76–81, 2016.
- [24] M. Vukobratović and B. Borovac, “Zero-moment point?thirty five years of its life,” *International Journal of Humanoid Robotics*, vol. 1, no. 01, pp. 157–173, 2004.
- [25] S. Kajita, T. Nagasaki, K. Kaneko, K. Yokoi, and K. Tanie, “A hop towards running humanoid biped,” vol. 1, pp. 629–635, 2004.
- [26] G. A. Cardona, W. Moreno, A. Weitzenfeld, and J. M. Calderon, “Reduction of impact force in falling robots using variable stiffness,” in *SoutheastCon 2016*. IEEE, 2016, pp. 1–6.
- [27] J. M. Calderón, W. Moreno, and A. Weitzenfeld, “Fuzzy variable stiffness in landing phase for jumping robot,” in *Innovations in Bio-Inspired Computing and Applications*. Springer, 2016, pp. 511–522.

- [28] Y. S. R. W. Chiaki Aoyama and S. M. N. H. K. Fujimura, "The intelligent asimo: System overview and integration."
- [29] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, "The nao humanoid: a combination of performance and affordability," *CoRR abs/0807.3223*, 2008.
- [30] I. Ha, Y. Tamura, H. Asama, J. Han, and D. W. Hong, "Development of open humanoid platform darwin-op," in *SICE Annual Conference (SICE), 2011 Proceedings of*. IEEE, 2011, pp. 2178–2181.
- [31] M. H. Raibert, H. B. Brown, and M. Chepponis, "Experiments in balance with a 3d one-legged hopping machine," *The International Journal of Robotics Research*, vol. 3, no. 2, pp. 75–92, 1984.
- [32] P. M. Wensing and D. E. Orin, "Control of humanoid hopping based on a slip model," in *Advances in Mechanisms, Robotics and Design Education and Research*. Springer, 2013, pp. 265–274.
- [33] —, "High-speed humanoid running through control with a 3d-slip model," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 5134–5140.
- [34] P. M. Wensing, "Optimization and control of dynamic humanoid running and jumping," Ph.D. dissertation, Citeseer, 2014.
- [35] S. Sakka and K. Yokoi, "Humanoid vertical jumping based on force feedback and inertial forces optimization," in *IEEE International Conference on Robotics and Automation*, vol. 4. IEEE; 1999, 2005, p. 3752.
- [36] S. Sakka, N. E. Sian, and K. Yokoi, "Motion pattern for the landing phase of a vertical jump for humanoid robots," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 5477–5483.
- [37] V. Nunez and N. Nadjar-Gauthier, "Humanoid vertical jump with compliant contact," in *Climbing and Walking Robots*. Springer, 2006, pp. 457–464.
- [38] D. Goswami and P. Vadakkepat, "Planar bipedal jumping gaits with stable landing," *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 1030–1046, 2009.
- [39] M. Missura and S. Behnke, "Self-stable omnidirectional walking with compliant joints," in *Proceedings of 8th Workshop on Humanoid Soccer Robots, IEEE Int. Conf. on Humanoid Robots, Atlanta, USA*, 2013.
- [40] Y. Liu, P. M. Wensing, D. E. Orin, and J. P. Schmiedeler, "Fuzzy controlled hopping in a biped robot," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1982–1989.
- [41] M. S. Hester, "Stable control of jumping in a planar biped robot," Ph.D. dissertation, The Ohio State University, 2009.
- [42] F. Daerden, "Conception and realization of pleated pneumatic artificial muscles and their use as compliant actuation elements," *Vrije Universiteit Brussel, Belgium*, 1999.




- [43] P. Beyl, B. Vanderborght, R. Van Ham, M. Van Damme, R. Versluys, and D. Lefeber, "Compliant actuation in new robotic applications," in *NCTAM06?7th National Congress on Theoretical and Applied Mechanics*, 2006.
- [44] J. Vermeulen, "Trajectory generation for planar hopping and walking robots: An objective parameter and angular momentum approach," *Vrije Universiteit Brussel, Brussel*, 2004.
- [45] B. Vanderborght, B. Verrelst, R. Van Ham, M. Van Damme, P. Beyl, and D. Lefeber, "Torque and compliance control of the pneumatic artificial muscles in the biped" lacy", in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, 2006, pp. 842–847.
- [46] J. Babič and J. Lenarčič, "Vertical jump: Biomechanical analysis and simulation study," *Humanoid Robots, New Developments*, p. 582, 2007.
- [47] J. Babič and J. Lenarčič, "Optimization of biarticular gastrocnemius muscle in humanoid jumping robot simulation," *International Journal of Humanoid Robotics*, vol. 3, no. 02, pp. 219–234, 2006.
- [48] E. Elibol, J. Calderon, and A. Weitzenfeld, "Optimizing energy usage through variable joint stiffness control during humanoid robot walking," in *Robot Soccer World Cup.* Springer, 2013, pp. 492–503.
- [49] J. M. Calderon, E. Elibol, W. Moreno, and A. Weitzenfeld, "Current usage reduction through stiffness control in humanoid robot," in *8th Workshop on Humanoid Soccer Robots, IEEE-RAS International Conference on Humanoid Robots*, 2013.
- [50] R. Renner and S. Behnke, "Instability detection and fall avoidance for a humanoid using attitude sensors and reflexes," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems.* IEEE, 2006, pp. 2967–2973.
- [51] J. D. Karssen and M. Wisse, "Fall detection in walking robots by multi-way principal component analysis," *Robotica*, vol. 27, no. 02, pp. 249–257, 2009.
- [52] K. Fujiwara, F. Kanehiro, S. Kajita, K. Kaneko, K. Yokoi, and H. Hirukawa, "Falling motion control to minimize damage to biped humanoid robot," in *Conference on Intelligent Robots and Systems EPFL, Lausanne, Switzerland*, 2002.
- [53] K. Fujiwara, F. Kanehiro, H. Saito, S. Kajita, K. Harada, and H. Hirukawa, "Falling motion control of a humanoid robot trained by virtual supplementary tests." in *IEEE International Conference on Robotics and Automation*, vol. 2. IEEE; 1999, 2004, pp. 1077–1082.
- [54] J. Ruiz-del Solar, J. Moya, and I. Parra-Tsunekawa, "Fall detection and management in biped humanoid robots," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on.* IEEE, 2010, pp. 3323–3328.
- [55] T. Wilken, M. Missura, and S. Behnke, "Designing falling motions for a humanoid soccer goalie," in *Proc. of the 4th Workshop on Humanoid Soccer Robots, Int. Conf. on Humanoid Robots*, 2009.
- [56] S. Ha, Y. Ye, and C. K. Liu, "Falling and landing motion control for character animation," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 155, 2012.


- [57] S. Ha and C. K. Liu, "Multiple contact planning for minimizing damage of humanoid falls," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 2761–2767.
- [58] L. Hunt, R. Su, and G. Meyer, "Global transformations of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 28, no. 1, pp. 24–31, 1983.
- [59] E. G. Gilbert and I. J. Ha, "An approach to nonlinear feedback control with applications to robotics," *IEEE transactions on systems, man, and cybernetics*, no. 6, pp. 879–884, 1984.
- [60] F. Piltan, M. H. Yarmahmoudi, M. Shamsodini, E. Mazlomian, and A. Hosainpour, "Puma-560 robot manipulator position computed torque control methods using matlab/simulink and their integration into graduate nonlinear control and matlab courses," *International Journal of Robotics and Automation*, vol. 3, no. 3, pp. 167–191, 2012.
- [61] S. Das, I. Pan, K. Halder, S. Das, and A. Gupta, "Lqr based improved discrete pid controller design via optimum selection of weighting matrices using fractional order integral performance index," *Applied Mathematical Modelling*, vol. 37, no. 6, pp. 4253–4268, 2013.
- [62] G.-R. Yu and R.-C. Hwang, "Optimal pid speed control of brush less dc motors using lqr approach," in *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, vol. 1. IEEE, 2004, pp. 473–478.
- [63] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International journal of man-machine studies*, vol. 7, no. 1, pp. 1–13, 1975.

APPENDICES

Appendix A: Permission for Reuse

The permissions below are for reuse of material in chapters 2, 3, 4, 5, and 6 respectively.

HomeCreate AccountHelpLive Chat

**IEEE**
Requesting permission to reuse content from an IEEE publication

Title: Reduction of impact force in falling robots using variable stiffness

Conference Proceedings: SoutheastCon, 2016

Author: Gustavo A. Cardona

Publisher: IEEE

Date: March 2016

Copyright © 2016, IEEE

LOGIN

If you're a **copyright.com** user, you can login to RightsLink using your copyright.com credentials. Already a **RightsLink** user or want to [learn more?](#)

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

- 1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
- 2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
- 3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

- 1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
- 2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
- 3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK

CLOSE WINDOW

Copyright © 2016 [Copyright Clearance Center, Inc.](#) All Rights Reserved. [Privacy statement.](#) [Terms and Conditions.](#)
Comments? We would like to hear from you. E-mail us at customercare@copyright.com

Appendix A (Continued)

SPRINGER LICENSE TERMS AND CONDITIONS

Oct 11, 2016

This Agreement between Juan M Calderon ("You") and Springer ("Springer") consists of your license details and the terms and conditions provided by Springer and Copyright Clearance Center.

License Number	3966140035624
License date	Oct 11, 2016
Licensed Content Publisher	Springer
Licensed Content Publication	Springer eBook
Licensed Content Title	Fuzzy Variable Stiffness in Landing Phase for Jumping Robot
Licensed Content Author	Juan M. Calderón
Licensed Content Date	Jan 1, 2016
Type of Use	Thesis/Dissertation
Portion	Full text
Number of copies	1
Author of this Springer article	Yes and you are the sole author of the new work
Order reference number	
Title of your thesis / dissertation	Impact Force Reduction Using Variable Stiffness With An Optimal Approach For Jumping and Falling Robots
Expected completion date	Dec 2016
Estimated size(pages)	80
Requestor Location	Juan M Calderon 2212 Village Park Road apt 302 PLANT CITY, FL 33563 United States Attn: Juan M Calderon
Billing Type	Invoice
Billing Address	Juan M Calderon 2212 Village Park Road apt 302 PLANT CITY, FL 33563 United States Attn: Juan M Calderon
Total	0.00 USD
Terms and Conditions	

Introduction

The publisher for this copyrighted material is Springer. By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the Billing and Payment terms and conditions

Appendix A (Continued)

established by Copyright Clearance Center, Inc. ("CCC"), at the time that you opened your Rightslink account and that are available at any time at <http://myaccount.copyright.com>.

Limited License

With reference to your request to reuse material on which Springer controls the copyright, permission is granted for the use indicated in your enquiry under the following conditions:

- Licenses are for one-time use only with a maximum distribution equal to the number stated in your request.

- Springer material represents original material which does not carry references to other sources. If the material in question appears with a credit to another source, this permission is not valid and authorization has to be obtained from the original copyright holder.

- This permission

- is non-exclusive

- is only valid if no personal rights, trademarks, or competitive products are infringed.

- explicitly excludes the right for derivatives.

- Springer does not supply original artwork or content.

- According to the format which you have selected, the following conditions apply accordingly:

- **Print and Electronic:** This License include use in electronic form provided it is password protected, on intranet, or CD-Rom/DVD or E-book/E-journal. It may not be republished in electronic open access.

- **Print:** This License excludes use in electronic form.

- **Electronic:** This License only pertains to use in electronic form provided it is password protected, on intranet, or CD-Rom/DVD or E-book/E-journal. It may not be republished in electronic open access.

For any electronic use not mentioned, please contact Springer at permissions.springer@spi-global.com.

- Although Springer controls the copyright to the material and is entitled to negotiate on rights, this license is only valid subject to courtesy information to the author (address is given in the article/chapter).

- If you are an STM Signatory or your work will be published by an STM Signatory and you are requesting to reuse figures/tables/illustrations or single text extracts, permission is granted according to STM Permissions Guidelines: <http://www.stm-assoc.org/permissions-guidelines/>

For any electronic use not mentioned in the Guidelines, please contact Springer at permissions.springer@spi-global.com. If you request to reuse more content than stipulated in the STM Permissions Guidelines, you will be charged a permission fee for the excess content.

Permission is valid upon payment of the fee as indicated in the licensing process. If permission is granted free of charge on this occasion, that does not prejudice any rights we might have to charge for reproduction of our copyrighted material in the future.

-If your request is for reuse in a Thesis, permission is granted free of charge under the following conditions:

This license is valid for one-time use only for the purpose of defending your thesis and with a maximum of 100 extra copies in paper. If the thesis is going to be published, permission needs to be reobtained.

- includes use in an electronic form, provided it is an author-created version of the thesis on his/her own website and his/her university's repository, including UMI (according to the definition on the Sherpa website: <http://www.sherpa.ac.uk/romeo/>);

- is subject to courtesy information to the co-author or corresponding author.

Geographic Rights: Scope

Licenses may be exercised anywhere in the world.

Altering/Modifying Material: Not Permitted

Figures, tables, and illustrations may be altered minimally to serve your work. You may not

Appendix A (Continued)

alter or modify text in any manner. Abbreviations, additions, deletions and/or any other alterations shall be made only with prior written authorization of the author(s).

Reservation of Rights

Springer reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction and (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.

License Contingent on Payment

While you may exercise the rights licensed immediately upon issuance of the license at the end of the licensing process for the transaction, provided that you have disclosed complete and accurate details of your proposed use, no license is finally effective unless and until full payment is received from you (either by Springer or by CCC) as provided in CCC's Billing and Payment terms and conditions. If full payment is not received by the date due, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment terms and conditions, the license is automatically revoked and shall be void as if never granted. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and Springer reserves the right to take any and all action to protect its copyright in the materials.

Copyright Notice: Disclaimer

You must include the following copyright and permission notice in connection with any reproduction of the licensed material:

"Springer book/journal title, chapter/article title, volume, year of publication, page, name(s) of author(s), (original copyright notice as given in the publication in which the material was originally published) "With permission of Springer"

In case of use of a graph or illustration, the caption of the graph or illustration must be included, as it is indicated in the original publication.

Warranties: None

Springer makes no representations or warranties with respect to the licensed material and adopts on its own behalf the limitations and disclaimers established by CCC on its behalf in its Billing and Payment terms and conditions for this licensing transaction.

Indemnity

You hereby indemnify and agree to hold harmless Springer and CCC, and their respective officers, directors, employees and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.

No Transfer of License

This license is personal to you and may not be sublicensed, assigned, or transferred by you without Springer's written permission.

No Amendment Except in Writing

This license may not be amended except in a writing signed by both parties (or, in the case of Springer, by CCC on Springer's behalf).

Objection to Contrary Terms

Springer hereby objects to any terms contained in any purchase order, acknowledgment, check endorsement or other writing prepared by you, which terms are inconsistent with these terms and conditions or CCC's Billing and Payment terms and conditions. These terms and conditions, together with CCC's Billing and Payment terms and conditions (which are incorporated herein), comprise the entire agreement between you and Springer (and CCC) concerning this licensing transaction. In the event of any conflict between your obligations established by these terms and conditions and those established by CCC's Billing and Payment terms and conditions, these terms and conditions shall control.

Jurisdiction

All disputes that may arise in connection with this present License, or the breach thereof,

Appendix A (Continued)

shall be settled exclusively by arbitration, to be held in the Federal Republic of Germany, in accordance with German law.

Other conditions:

V 12AUG2015

Questions? customercare@copyright.com or +1-855-239-3415 (toll free in the US) or +1-978-646-2777.
