論文 / 著書情報
Article / Book Information

題目(和文)	
Title(English)	Robust and Efficient Bipedal Walking Robot Control via Trajectory Optimization and Biarticular Muscles
著者(和文)	PELIT Mustafa Melih
Author(English)	Mustafa Melih Pelit
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第12535号, 授与年月日:2023年9月22日, 学位の種別:課程博士, 審査員:山北 昌毅,三平 満司,倉林 大輔,塚越 秀行,畑中 健志
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第12535号, Conferred date:2023/9/22, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

Robust and Efficient Bipedal Walking Robot Control via Trajectory Optimization and Biarticular Muscles

A thesis presented

by

Mustafa Melih PELIT

Doctor of Engineering
in
Systems and Control Engineering

Supervisor

Associate Professor Masaki YAMAKITA

Department of Systems and Control Engineering
School of Engineering
Tokyo Institute of Technology

August, 2023

Abstract

Bipedal robots have the potential to significantly impact our lives by offering mobility capabilities beyond our own. This thesis investigates the utilization of template models, trajectory optimization techniques, biarticular muscles and various control techniques to achieve walking gait of a 5-link bipedal robot through simulation studies. The well- known spring-loaded inverted pendulum (SLIP) model is expanded to incorporate swing leg dynamics (SLIP-SL). Furthermore, the model is enhanced through the integration of variable stiffness control (VSLIP-SL) to improve its robustness against external disturbances. Additionally, the study demonstrates the effectiveness of passive biarticular muscles, an active wobbling mass and the combination of the two in enhancing the robustness and efficiency of bipedal robots. We propose a method that improves the terrain-blind walking on rough terrain and further improve the performance by adding and tuning biarticular springs. Finally, the VSLIP-SL model is employed to achieve gait patterns incorporating variable stiffness biarticular muscles. We propose various controllers to achieve stable walking gaits based on template models or trajectories obtained via trajectory optimization methods.

Key Words: legged robots, robot dynamics and control, bipedal robots, biarticular muscles, variable stiffness, terrain-blind walking, template walking models

Acknowledgements

I would like to express my deepest gratitude to my supervisor Professor Masaki Yamakita for his invaluable support and guidance. We first met in 2015 when I joined the Young Scientist Exchange Program in Tokyo Institute of Technology. With his advice, support and mentoring, I was able to conduct my studies with excitement and inspiration since then. I would like to thank committee members Professor Hideyuki Tsukagoshi, Professor Daisuke Kurabayashi, Professor Takeshi Hatanaka and Professor Mitsuji Sampei, who kindly spent time checking and approving this thesis. My thanks extend to Professor Ali Emre Turgut whom I conducted research with and learned a lot from during my bachelors study.

I would like to use this opportunity to truly thank Dr. Junho Chang and Dr. Rin Takano who extended their kind help whenever I needed it throughout my research. I thank former and current Yamakita Lab members for creating this friendly and warm research environment. I thank Dr. Dmitrii Koldaev, Kyu Hur, Jason Wang, Vatan Tezer, Dr. Gajamohan Mohanarajah and Christof Dubs from Rapyuta Robotics for their support. I would like to express my gratitude to my friends Emek Barış Küçüktabak, Doğa Can Atabay, Berke Aral Sönmez, Arda Lüleci, Mehmet Yıldız, Berkan Bilir, Haris Hasic. I am especially thankful to Tutku Turgay who has been on my side since the beginning of my bachelors to this day, guiding and supporting me.

Last but not least, I would like to sincerely thank my family who always supported me and took care of me. I am truly grateful for my parents Huseyin Avni Pelit, Aysel Pelit and my brother Mehmet Pelit for their love and support.

Mustafa Melih PELIT
August 2023, Tokyo

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Robots have become increasingly prevalent in various aspects of our lives. From industrial settings where they assist in manufacturing processes to domestic environments where they automate household chores such as vacuuming, robots have permeated numerous fields and applications. While some robots are stationary, mobility plays a pivotal role in the majority of robotics applications.

The ability of a robot to move detached from any base provides significant advantages and expands its range of capabilities. Mobile robots possess the potential to navigate and operate in different environments, thereby extending their utility and versatility. By enabling movement, robots can access diverse locations, interact with objects and surroundings, and perform tasks that require locomotion.

The desire for mobile robots is rooted in the observation that mobility is a fundamental characteristic of many living organisms. In nature, organisms with the ability to move exhibit a distinct advantage in terms of survival, exploration, and adaptation. Inspired by this principle, the development of mobile robots seeks to emulate the benefits conferred by movement, enabling robots to navigate and interact with their surroundings in a manner that aligns with the behavior of living organisms.

The ability to move is undeniably valuable, prompting the question of how robots should navigate their environments. There are numerous options to consider when determining the most suitable locomotion method. Wheels, for instance, are well-suited for certain tasks due to their stable and controllable movement on flat surfaces. However, they struggle with discontinuous ground support like when climbing stairs. On the other hand, quad rotors or drones provide the capability of flight but are vulnerable to significant damage from even minor crashes. These examples merely scratch the surface, as researchers employ a range of intriguing methods to construct mobile robots. Among these techniques, two-legged locomotion stands out as particularly captivating. It closely mimics human movement and offers a host of advantages, making it a compelling choice for various applications.

Bipedal locomotion offers several distinct advantages in robotics. Firstly, it enables robots to maneuver effectively in narrow spaces, thanks to their ability to turn and navigate with

agility. Additionally, bipedal robots excel in traversing terrains with uneven or discontinuous ground support, such as climbing stairs or navigating rocky slopes. Their versatile locomotion capabilities allow for seamless transitions between different modes, including walking, running, and even jumping, depending on the situational requirements.

One of the most significant advantages of bipedal robots is their compatibility with environments designed for humans. By emulating human-like movement, bipedal robots are well-suited for operating in environments such as offices, factories, and hospitals. This compatibility enhances their adaptability and facilitates seamless integration into human-centric spaces.

In this thesis, we try to improve some aspects of bipedal walking in the hopes of contributing to the process of having efficient, robust and fast bipedal robots help with dull, difficult and dangerous jobs so we as humans can focus on more enjoyable and creative work. We test our proposed methods on simulation studies. Our test bed is a 5-link bipedal robot model that is constrained to the sagittal plane.

This work has 4 main contributions:

- Developing a simple yet robust template model for walking (Chapters 2 and 3)
- Investigating the effects of introducing biarticular compliant elements and a wobbling mass on the walking performance through trajectory optimization studies (Chapters 4 and 5)
- Proposing a controller and bipedal robot design that improves the terrain-blind walking performance (Chapter 6)
- Proposing a controller that can control variable stiffness biarticular springs to increase robustness (Chapter 7)
- Coming up with controllers that can achieve efficient/robust walking, using the template models or optimal trajectories that were developed in this study (all the chapters except Chapter 5)

1.1 Outline

This thesis consists of the below chapters, and its contents are organized as follows.

Chapter 2 Extending Spring Loaded Inverted Pendulum (SLIP) with Swing Leg Dynamics (SLIP-SL)

This chapter details an extension to the SLIP model named spring loaded inverted pendulum model with swing legs (SLIP-SL). SLIP-SL extends the SLIP model by introducing swing leg dynamics while keeping its passive nature. This way, reference trajectories for the center of mass and swing foot trajectories can be simultaneously obtained which was

not possible with the SLIP. This makes implementation easier and it increases tracking performance. We show how a variety of feasible two-phased walking trajectories can be obtained for this template model using direct collocation optimization methods. It is also shown through simulation studies that reference SLIP-SL trajectories can be used to control a fully actuated bipedal robot with the proposed feedback linearization controller to reach a stable cyclic gait.

Chapter 3 Variable Stiffness Spring Loaded Inverted Pendulum Model with Swing Leg Dynamics (VSLIP-SL)

The robustness issue of the SLIP was addressed by swapping the leg springs with variable stiffness ones in the variable SLIP (V-SLIP) model. The SLIP model with swing leg (SLIP-SL) introduced swing leg dynamics to the SLIP by adding a massed leg and foot, and springs to move them. In this chapter, we first propose a template model that combines the swing leg dynamics and robustness solutions in a new model called "the variable stiffness SLIP model with swing leg dynamics" (VSLIP-SL). Then, we go on to propose a controller for a 5-link fully actuated bipedal robot model that is based on the VSLIP-SL. The challenge of the controller is to translate the variable stiffness responses of the VSLIP-SL to the 5-link model, which does not have any compliant elements. We achieve this through an encoder-decoder scheme and feedback linearization. It is shown by numerical simulations that we are able to tackle the robustness issue on a template level.

Chapter 4 Effects of Passive Biarticular Muscles on Walking Performance for Bipedal Robots

The first goal of this chapter is to investigate the effects of passive biarticular muscles on the walking performance of a bipedal robot. Second goal is to achieve a stable walking gait with the bipedal robot model that has these biarticular muscles. To achieve this, we first compare the optimal trajectories of a default bipedal robot model with the same model that also has biarticular springs. The comparison was made with respect to walking efficiency, speed and minimum input torque requirements criteria using direct collocation methods. We show that the model with biarticular muscles outperforms the default model in all the investigated criteria and also proper parameters for the biarticular springs can be chosen using the described optimization method. Then, we introduce a feedback+feedforward controller to track the obtained optimal trajectories with a bipedal robot with biarticular muscles. Through simulations, we show that using the proposed controller, stable walking gaits can be achieved with good tracking performances.

Chapter 5 Effects of Active Wobbling Mass on Biped Robot's Walking Performance in Combination with Biarticular Springs

In other studies, it was shown that having an active wobbling mass (reaction mass) that can move up and down inside torso can increase the walking performance such as average walking velocity. In the previous chapter, we show that having biarticular springs that are tuned via the simultaneous trajectory-parameter optimization process can improve the

walking speeds too. In this chapter, we show that we can achieve faster walking speeds by combining the passive biarticular springs with an active wobbling mass. This combined model outperforms the nominal model, the model that has only biarticular muscles and the model that only has a wobbling mass.

Chapter 6 Terrain-Blind Humanoid Walking on Rough Terrain with Trajectory Optimization and Biarticular Springs

Trajectory optimization techniques to control biped walkers are becoming popular with improvements in available solvers. However, many of the proposed controllers assume that the terrain is flat, causing the biped robot to easily fall when the assumption doesn't hold. Humans can easily walk on rough terrain and there are a number of controllers that deal with this issue through perception or sensing but necessary research to tackle this issue without perception/sensing is still lacking. If the walking controller can deal with terrain changes without perception/sensing (terrain-blind), this would ease the computational burden on the controller and decrease the problems caused by errors in perception. This chapter proposes a controller that can track the optimized trajectories while handling moderate changes in terrain height. This was mainly achieved by our phase variable manipulation and utilization of a second optimized trajectory that lands the robot safely. We have also improved the robustness of the robot mechanically, by adding passive biarticular muscles. Furthermore, we investigated the effect of biarticular muscle parameters on robustness. Through simulation studies, we show that our proposed controller with proper biarticular muscle parameters can have a 5-link underactuated robot walk without falling on terrains with up to 6.47 cm height changes.

Chapter 7 Walking Control of a 5-Link Underactuated Bipedal Robot with Variable Stiffness Biarticular Springs Based VSLIP-SL

In Chapter 3, we've shown the effectiveness of the variable stiffness spring loaded inverted pendulum model swing leg dynamics (VSLIP-SL) model. In Chapter 4, we have shown that introduction of biarticular springs can significantly improve the walking performance. In this chapter, we model a 5-link model that has biarticular springs whose stiffness can be controlled. We propose a controller to achieve trajectory tracking while controlling the spring stiffnesses. This controller is based on the variable stiffness responses of VSLIP-SL. We show that this model can overcome external pushes thanks to the proposed controller scheme.

Chapter 2

Extending Spring Loaded Inverted Pendulum (SLIP) with Swing Leg Dynamics (SLIP-SL)

A plethora of techniques exist for controlling the locomotion of bipedal robots, offering researchers a diverse array of options. One extensively utilized approach involves employing an inverted pendulum model, augmented by the zero moment point (ZMP) criterion as the template model[1][2]. Furthermore, a new method gaining traction revolves around the derivation of optimal trajectories and inputs using various optimization techniques, subsequently employing them as reference values [3][4]. Additionally, researchers frequently employ simplistic models capable of mimicking key elements of human or animal gait, serving as blueprint models for bipedal robot walking control. Among these template models, the bipedal spring-loaded inverted pendulum (SLIP) model stands out as a widely embraced choice [5][6]. The utilization of these distinct methodologies showcases the breadth and depth of research endeavors aimed at advancing the control strategies employed in bipedal robotics.

The Bipedal Spring-Loaded Inverted Pendulum (SLIP) model, illustrated in Figure 2.3, comprises compliant legs and a point mass. This model operates passively, meaning it lacks external inputs, and its motion is governed solely by mechanical parameters and initial conditions. By carefully selecting these factors, a variety of trajectories can be achieved, converging into a repetitive pattern known as a limit cycle. Notably, the Bipedal SLIP model is capable of emulating the two-phased walking style observed in humans, as extensively detailed in [7]. These two phases are called single stance phase and double stance phase. In the single stance phase, only one leg is in contact with the ground and the other leg is “swinging". And in the double stance phase, both feet are in contact with the ground.

Although the Bipedal Spring-Loaded Inverted Pendulum (SLIP) model is highly advantageous and simplistic, it incorporates a significant assumption that deviates substantially from human gait. Specifically, in the single stance phase, the swing leg is assumed to instantaneously move to the appropriate touch-down position. Consequently, when utilizing the bipedal SLIP model as a template for controlling a bipedal robot, it is unable to provide

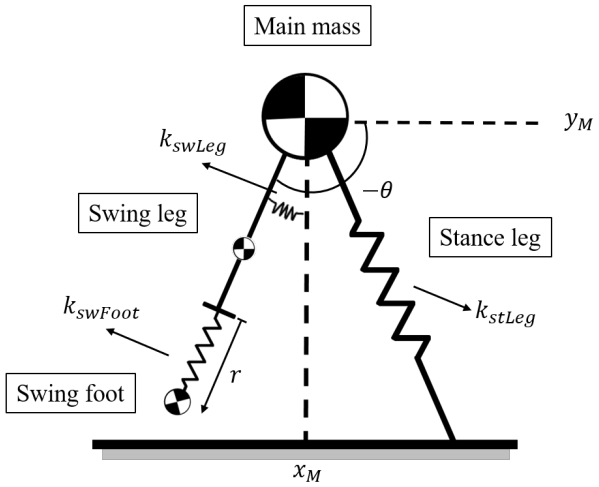


Figure 2.1: SLIP-SL Model in the single stance phase

the desired trajectories for the swing foot. Additional measures are therefore necessary to achieve the motion of the swing leg. In an effort to address this limitation, Sharbafi et al. proposed a conceptual model in [8] by combining the SLIP model with a segmented leg. This hybrid model enables manages to incorporate the swinging motion, but it requires the implementation of reference trajectories and a controller to achieve the desired motion. Notably, the introduction of these control elements results in the model losing its passive characteristics.

In this chapter, we propose an extended SLIP model from which reference center of mass trajectories and swing foot trajectories can be obtained simultaneously. This model will be called spring loaded inverted pendulum model with swing leg dynamics (SLIP-SL) which is shown in Figure 2.1 and it will keep the passive property of the SLIP model.

SLIP-SL (Spring-Loaded Inverted Pendulum with Segmented Legs) model, also exhibits a two-phase gait similar to the bipedal SLIP model. In the single stance phase, the SLIP-SL model comprises of three massed elements and three springs to facilitate the movement of these components. Importantly, the SLIP-SL model retains the passive nature of the bipedal SLIP since it does not require any external inputs. In other words, by adding two massed elements and two springs to the original SLIP model, the SLIP-SL model maintains its passive characteristics. To ensure feasible trajectories, it is essential to carefully select appropriate spring parameters and initial conditions such as position and velocity. To this end, we employ direct collocation methods [9] to simultaneously optimize the model's parameters and trajectory. Furthermore, we propose a feedback linearization controller to track the obtained SLIP-SL trajectories using a 5-link fully actuated bipedal robot model. Through simulation studies, we investigate the effectiveness of the proposed controller and examine the SLIP-SL model's suitability as a template for walking.

This chapter is organized as follows: Section 2.1 describes the dynamics of SLIP-SL and the bipedal robot model, Section 2.2 details the optimization setup that is needed for finding

feasible SLIP-SL trajectories, Section 2.3 introduces the proposed feedback linearization controller and in Section 2.4 simulation results are presented and discussed.

2.1 Systems and Modeling

This section will begin by introducing the original bipedal SLIP model. Then, we introduce the dynamic model of the extended SLIP model named SLIP-SL. Finally the model for the bipedal robot will be introduced. Explanation of the SLIP model will be brief since there are many works such as [5] that do an excellent job and going in depth on the matter. This chapter will focus on the extended model and fully covers it but it is recommended to have a basic knowledge of the SLIP model to be able to appreciate the extension of it.

2.1.1 Bipedal SLIP Model

The Bipedal Spring-Loaded Inverted Pendulum (SLIP) Model, depicted in Figure 2.3, is comprised of a point mass and two legs made of springs with negligible mass. This model emulates the two distinct phases observed in human walking, known as the single stance and double stance phases. During the double stance phase, both feet are in contact with the ground, and the spring-like legs exert forces on the point mass. The transition from the single stance to the double stance phase occurs when the lift-off event is triggered. Lift-off transpires when the trailing foot disengages from the ground, initiating the single stance phase. This phase continues until the touch-down event, which transpires when the swinging foot makes contact with the ground. The configuration of the touch-down depends on the angle of attack α and the natural length of the spring L_0. Following the touch-down event, the SLIP model switches back to the double stance phase, and this cyclical process repeats to achieve a walking motion, as illustrated in Figure 2.2. It is important to note that the springs in the model can only contract and exert forces, as a leg in reality cannot pull us towards the ground.

The Bipedal Spring-Loaded Inverted Pendulum (SLIP) model is a passive system, meaning its motion is solely determined by its parameters, such as spring stiffness, angle of attack, and initial conditions, including initial velocity. By adjusting these parameters and initial conditions, various types of gaits can be achieved with this model, allowing for the generation of stable, human-like walking patterns characterized by double-peaked ground reaction forces. The simplicity of the SLIP model has made it a popular choice among researchers,

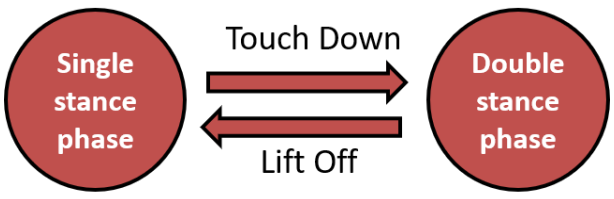


Figure 2.2: SLIP, SLIP-SL and 5-link biped robot model phase transition diagram

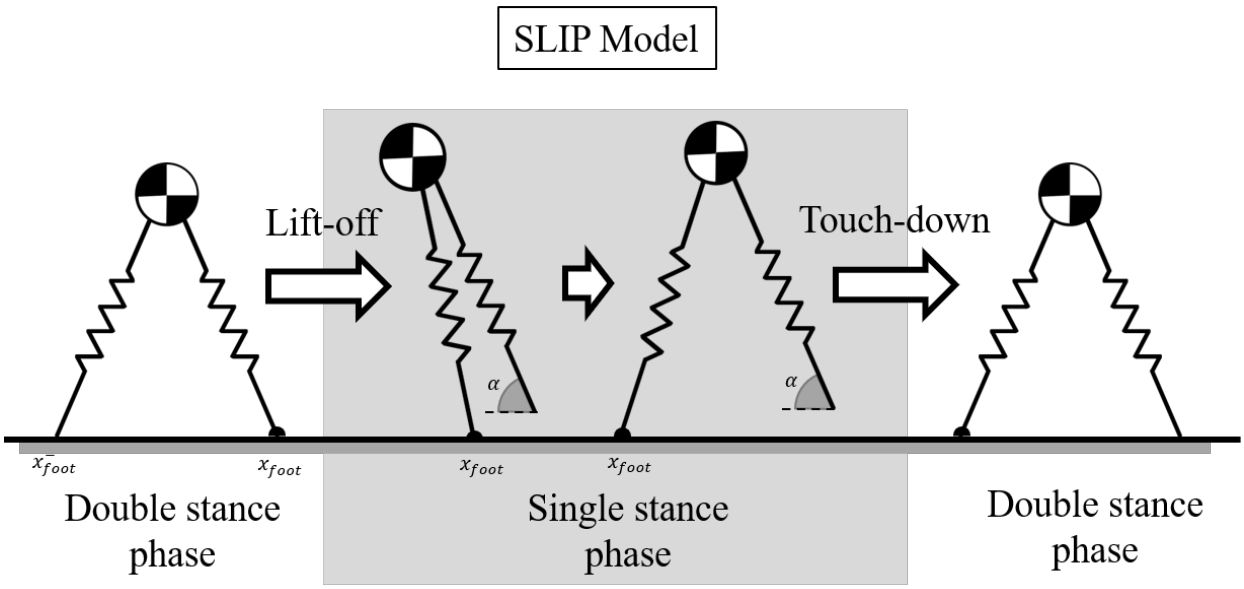


Figure 2.3: Bipedal SLIP Model

particularly for generating reference trajectories for the center of mass (CoM) position.

However, the SLIP model makes a significant assumption that the swing leg can instantaneously move to the required position for touch-down. This assumption is feasible for the model because its legs are considered to be massless. Nevertheless, for real robots, the swing leg motion needs to be physically moved in order to facilitate walking. Consequently, when the SLIP model is used as a reference, additional steps are necessary to generate the motion of the swing leg. Additionally, incorporating the swing leg motion on biped robot that follows the SLIP model CoM trajectory can prove challenging and can even potentially act as a disturbance.

To address these limitations, we propose the SLIP-SL model, which extends the SLIP model by incorporating the dynamics of the swing leg during the single stance phase. This extension allows for a more comprehensive representation of the walking process, capturing the essential swing leg motion alongside the CoM trajectory.

2.1.2 SLIP-SL model

The Spring-Loaded Inverted Pendulum model with Swing Leg (SLIP-SL), illustrated in Figure 2.1, exhibits a walking motion that can be visualized in Figure 2.4 over the course of a complete step. Similar to its predecessor, the SLIP model, SLIP-SL operates in two distinct phases. During the double stance phase, both SLIP-SL and SLIP models are identical in their behavior.

However, what sets the SLIP-SL model apart from the SLIP model is the incorporation

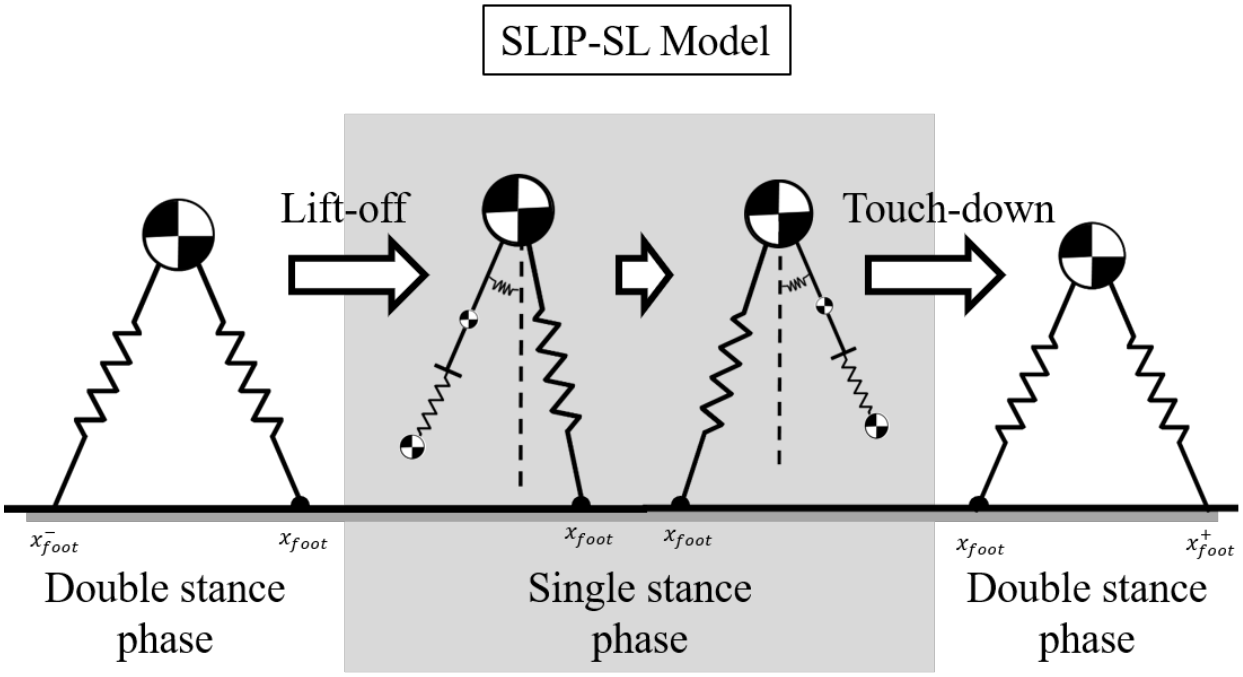


Figure 2.4: SLIP-SL Model

of swing leg dynamics during the single stance phase. This addition allows the SLIP-SL model to account for the motion and behavior of the swing leg as it plays a crucial role in generating a more realistic walking motion.

During the single stance phase, the SLIP-SL model consists of three massed elements: the primary mass denoted as M, a massed swing leg, and a point mass representing the swing foot. Additionally, there are three springs involved: a linear spring connecting M' to the ground, a torsional spring connecting the point mass to the swing leg, and a linear spring connecting the swing foot to the swing leg. This configuration allows the SLIP-SL model to move these elements during the single stance phase.

After the touch-down event, when the swing foot makes contact with the ground, the SLIP-SL model transitions to the double stance phase. In this phase, the SLIP-SL model in the double stance phase consists of a point mass and two massless springs, resembling the structure of the SLIP model during this phase.

In the single stance phase, equation of motion for the SLIP-SL model can be written as:

$$\tilde{\boldsymbol{M}}(\tilde{\boldsymbol{q}})\ddot{\tilde{\boldsymbol{q}}} + \tilde{\boldsymbol{C}}(\tilde{\boldsymbol{q}}, \dot{\tilde{\boldsymbol{q}}}) + \tilde{\boldsymbol{G}}(\tilde{\boldsymbol{q}}) = \tilde{\boldsymbol{S}}\tilde{\boldsymbol{\tau}} \tag{2.1}$$

where $\tilde{\boldsymbol{q}} = [x_M, y_M, \theta, r]^T$ are the generalized coordinates, $\tilde{\boldsymbol{M}}(\tilde{\boldsymbol{q}}) \in \mathbb{R}^{4\times4}$ is an inertia matrix, $\tilde{\boldsymbol{C}}(\tilde{\boldsymbol{q}}, \dot{\tilde{\boldsymbol{q}}}) \in \mathbb{R}^4$ is a Coriolis and centrifugal terms vector, $\tilde{\boldsymbol{G}}(\tilde{\boldsymbol{q}}) \in \mathbb{R}^4$ is the gravity term, $\tilde{\boldsymbol{\tau}} \in \mathbb{R}^3$ are the resultant forces and torques due to springs and $\tilde{\boldsymbol{S}} \in \mathbb{R}^{4\times3}$ is the appropriate mapping matrix for them. These can be calculated as:

$$\tilde{\boldsymbol{\tau}} = \begin{bmatrix} k_{0,\mathrm{ss}}(L_{0,ss} - L_{st,ss}) \\ k_{\mathrm{swLeg}}(\theta_0 - \theta) \\ k_{\mathrm{swFoot}}(r_0 - r) \end{bmatrix}, \tag{2.2}$$

where $k_{0,\mathrm{ss}}$, k_{swLeg} and k_{swFoot} are the stiffness values for the stance leg spring, the torsional spring at the "hip" and the linear spring connecting the swing foot with the swing leg, respectively and $L_{0,\mathrm{ss}}$, θ_0 and r_0 are the free positions of those springs where subscript "ss" indicates the single stance phase. L_{st} is the length of the stance leg. x_M and y_M respectively represent horizontal and vertical positions of the main mass, θ represents the angle of the swing leg with respect to the vertical axis and r represents the distance between the end of the swing leg and swing foot point which are represented in Figure 2.1.

In the double stance phase, dynamics of the SLIP-SL model can be written as:

$$m \begin{bmatrix} \ddot{x}_{\mathrm{CoM}} \\ \ddot{y}_{\mathrm{CoM}} \end{bmatrix} = \boldsymbol{F}_{\mathrm{sw}} + \boldsymbol{F}_{\mathrm{st}} + m\boldsymbol{g}, \tag{2.3}$$

where

$$m = m_M + m_{\mathrm{swLeg}} + m_{\mathrm{swFoot}}, \tag{2.4}$$

indicates the total mass of the system, x_{CoM} and y_{CoM} are the horizontal and vertical positions of the center of mass and $\boldsymbol{g} = [0, -9.81]^T$ is the gravitational acceleration. Forces generated by the stance and swing leg springs can be calculated as:

$$\boldsymbol{F}_{st} = k_{0,\mathrm{ds}} \left(\frac{L_{0,\mathrm{ds}}}{L_{st}} - 1 \right) \left(\begin{bmatrix} x_{\mathrm{CoM}} \\ y_{\mathrm{CoM}} \end{bmatrix} - \begin{bmatrix} x_{foot} \\ 0 \end{bmatrix} \right), \tag{2.5}$$

$$\boldsymbol{F}_{sw} = k_{0,\mathrm{ds}} \left(\frac{L_{0,\mathrm{ds}}}{L_{sw}} - 1 \right) \left(\begin{bmatrix} x_{\mathrm{CoM}} \\ y_{\mathrm{CoM}} \end{bmatrix} - \begin{bmatrix} x^{-}_{foot} \\ 0 \end{bmatrix} \right), \tag{2.6}$$

where subscript "ds" indicates the double stance phase. Definitions of x_{foot} can be seen in Figure 2.4.

2.1.3 Bipedal Robot Model

In this part, the dynamics of the 5-link bipedal robot will be introduced. This model consists of 5 links which are connected to each other with revolute joints and it moves in the sagittal plane. The model is fully actuated and has an ankle torque.

Dynamics of the bipedal robot in the single stance phase can be written as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{S}\boldsymbol{u}, \tag{2.7}$$

where $\boldsymbol{q} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^T \in \mathbb{R}^5$ are the generalized coordinates, $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{5\times5}$ is an inertia matrix, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{5\times5}$ is a Coriolis and centrifugal terms matrix, $\boldsymbol{G}(\boldsymbol{q}) \in \mathbb{R}^5$ is the gravity term, $\boldsymbol{S} \in \mathbb{R}^{5\times5}$ is the distribution matrix of actuation torques and $\boldsymbol{u} \in \mathbb{R}^5$ are the input torques. This model can be seen in Figure 2.5 with the description of generalized

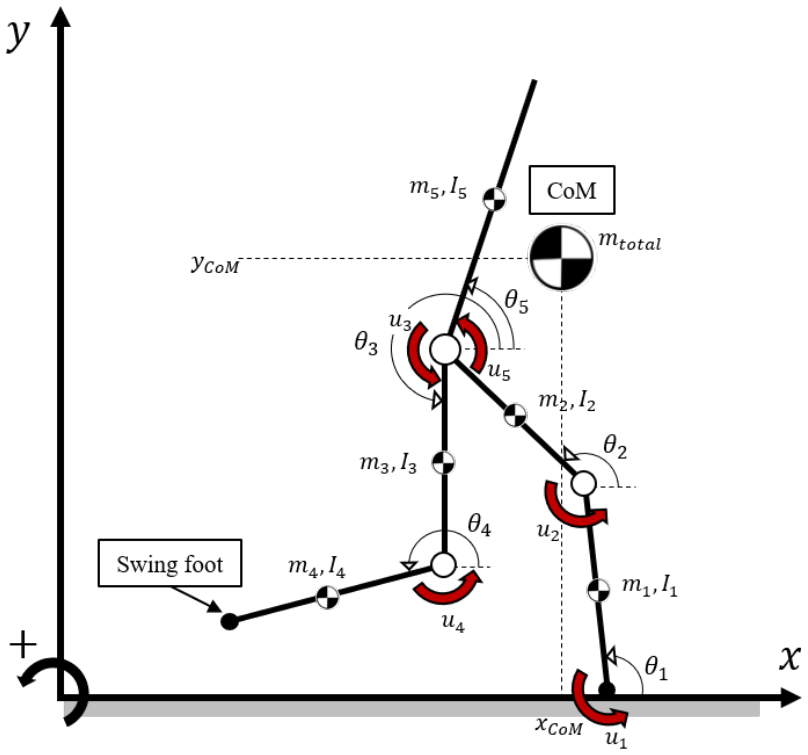


Figure 2.5: 5 link fully actuated robot model

coordinates and input torques (indicated with the red arrows).

Like the SLIP-SL model, bipedal robot model also has a two phased walking pattern. In the single stance phase, only one foot is on the ground and other is doing the swinging motion. Single stance phase ends and the system goes into the double stance phase when the swing foot touches the ground.

When the swing foot contacts the ground, a collision occurs where the generalized momentum of the system changes discontinuously. This can be modeled by assuming that an impulse force acts on the system to change the velocities while position is kept the same. This can be expressed as:

$$\boldsymbol{M}(\boldsymbol{q})\Delta\dot{\boldsymbol{q}} = \boldsymbol{J}_c^T \boldsymbol{\lambda}_{\text{impact}}, \tag{2.8}$$

where $\boldsymbol{J}_c \in \mathbb{R}^{2\times 5}$ is a constraint Jacobian matrix that maps the joint velocities to the swing foot velocity in horizontal and vertical directions. The generalized reaction forces in x and y directions are indicated as $\boldsymbol{\lambda}_{\text{impact}} = [\lambda^x_{\text{impact}}, \lambda^y_{\text{impact}}]^T \in \mathbb{R}^2$. Assuming that the impact is inelastic, velocity of the swing foot touching the ground will become zero after the impact which can be written as:

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}^+ = 0 \Leftrightarrow \boldsymbol{J}_c(\boldsymbol{q})\Delta\dot{\boldsymbol{q}} = -\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}^-, \tag{2.9}$$

where $^-$ superscript indicates the moment just before the impact and $^+$ just after the impact.

By solving (2.8) and (2.9) for $\boldsymbol{\lambda}_{impact}$ the following expression can be derived:

$$\boldsymbol{\lambda}_{\text{impact}} = -(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}\boldsymbol{J}_c\dot{\boldsymbol{q}}^- . \tag{2.10}$$

By inserting $\boldsymbol{\lambda}_{\text{impact}}$ into (2.8), we can obtain:

$$\dot{\boldsymbol{q}}^+ = (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{J}_c^T(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}\boldsymbol{J}_c)\dot{\boldsymbol{q}}^-, \tag{2.11}$$

which are the generalized velocities just after the impact. At the moment of impact, definitions of the legs are also switched (swing leg becomes the stance leg and vice versa).

The system is now in the double stance phase where both feet are in contact with the ground. To model this, a constraint force $\boldsymbol{\lambda}_{\text{ds}} = [\lambda_{\text{ds}}^x, \lambda_{\text{ds}}^y]^T \in \mathbb{R}^2$ is added to keep the swing foot on the ground. In the vertical direction, this constraint force can only push the robot ($\lambda_{ds}^y > 0$). With the non-slip assumption, double stance phase dynamics can be modeled by introducing the following constraint:

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0. \tag{2.12}$$

Using this constraint, dynamical equation of the double stance phase can be written as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{S}\boldsymbol{u} + \boldsymbol{J}_c^T\boldsymbol{\lambda}_{\text{ds}}. \tag{2.13}$$

The constraint force $\boldsymbol{\lambda}_{\text{ds}}$ can be obtained by taking time derivative of equation (2.12) as:

$$\boldsymbol{J}_c\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}_c\dot{\boldsymbol{q}} = 0, \tag{2.14}$$

and inserting it into (2.13) as follows:

$$\boldsymbol{\lambda}_{\text{ds}} = -(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}(\boldsymbol{J}_c\boldsymbol{M}^{-1}(\boldsymbol{S}\boldsymbol{u} - \boldsymbol{C}\dot{\boldsymbol{q}} - \boldsymbol{G}) + \dot{\boldsymbol{J}}_c\dot{\boldsymbol{q}}). \tag{2.15}$$

2.2 Direct Collocation Optimization

In this section, we will discuss the optimization process employed to determine periodic trajectories for the SLIP-SL model. As a passive model, the gait of SLIP-SL is entirely determined by its mechanical parameters and initial conditions. To identify suitable parameter values, Direct Collocation Methods [9] were employed.

Direct Collocation Methods handle the trajectory optimization problem by transforming them into a format compatible with nonlinear programming (NLP) solvers. This is primarily achieved through discretization, where the trajectory is divided into discrete segments. By optimizing these segments, an overall optimal trajectory can be obtained. There are several commercially available NLP solvers that efficiently handle these optimization problems.

The Direct Collocation Methods enabled the determination of periodic trajectories for the SLIP-SL model, finding the proper parameter values that produce desired walking patterns.

In this study, OpenOCL (Open Optimal Control Library) [10] will be used to solve the trajectory optimization problem. This solver can handle the multi-phase trajectory optimization problem. It can also optimize the chosen parameters while finding an optimal trajectory. This is called simultaneous parameter-trajectory optimization.

Finding the global minimum is not guaranteed when using the direct collocation methods. Global minimum is not easily found in a nonlinear problem with constraints such as 2.16. The advantage of direct collocation over other optimization methods such as genetic algorithms or learning based algorithms is that dynamics of the system can be embedded as constraints to the optimization problem painlessly.

The optimization problem can be formulated as:

$$\begin{aligned} \min_{\boldsymbol{x}_i, \boldsymbol{p}, T_i} \quad & \sum_{i=1}^{2} \left(\int_{T_{i-1}}^{T_i} J_i(\boldsymbol{x}_i(t), \boldsymbol{p}) dt \right) \text{ for } i \in \{1, 2\} \\ \text{s.t.} \quad & \dot{\boldsymbol{x}}_i = \boldsymbol{f}_i(\boldsymbol{x}_i(t), \boldsymbol{p}) \\ & \boldsymbol{r}_{i,k}(\boldsymbol{x}_i(\mu_{i,k}), \boldsymbol{p}) \leq 0, \end{aligned} \tag{2.16}$$

where $t \in [0, T_i]$ is the time, T_i is the end time of the respective phase, $\boldsymbol{x}_i(t)$ is the state trajectory, $\boldsymbol{p}$ are the parameters, $J_i(x, p)$ are the path cost functions, $\boldsymbol{f}_i(\boldsymbol{x}, \boldsymbol{p})$ are the system dynamics (described in Section 2.1.2) and $\boldsymbol{r}_{i,k}(\boldsymbol{x}_i, \boldsymbol{p})$ are the grid-constraints. $i = 1$ represents the single stance phase and $i = 2$ represents the double stance phase for SLIP-SL (T_1 is when the touch-down happens at the end of "ss" and T_2 is when the lift-off happens at the end of "ds"). In this chapter, the Cost of Transport (CoT) [5] and a cost function to keep the swing foot close to the ground was used. CoT is an indicator of the walking efficiency and efficiency is increased as CoT becomes smaller.

We constrain the system with path, boundary and stage transition constraints such that the solver finds feasible walking trajectories. Some of the constraints makes sure the resulting gait is human like while others makes it realistic such as joint angular acceleration limits. Path constraints are:

- Bounds were set for the parameters to be optimized:

$$\begin{aligned} & 15000 \leq k_{0,i} \leq 16000 \; [N/m], i \in \{ss, ds\} \\ & 1 \leq L_{0,\text{ss}} \leq 1.2 \; [m] \\ & 0 \leq k_{\text{swFoot}} \leq 20000 \; [N/m] \\ & 0 \leq k_{\text{swLeg}} \leq 15000 \; [Nm/rad] \\ & 0 \leq \theta_0 \leq 2\pi \; [rad] \\ & -10 \leq r_0 \leq 10 \; [m] \end{aligned} \tag{2.17}$$

- Stance leg spring in the single stance phase and both legs' springs in the double stance phase are always under contraction:

$$\begin{aligned} L_{st,ss} &\leq L_{0,ss} \\ L_{st,ds} &\leq L_{0,ds} \\ L_{sw,ds} &\leq L_{0,ds} \end{aligned} \tag{2.18}$$

- Constraining the vertical position of CoM:

$$0\ [m] \leq y_{\text{CoM}} \leq 0.85\ [m] \tag{2.19}$$

- Swing foot is always above the ground during the single stance phase:

$$y_{\text{swFoot}} \geq 0 \tag{2.20}$$

- Elliptic virtual obstacle must be avoided by the swing foot during the single stance phase:

$$\left(\frac{x_{\text{swFoot}} - d_{obs}}{w_{obs}}\right)^2 + \left(\frac{y_{\text{swFoot}}}{h_{obs}}\right)^2 \geq 1, \tag{2.21}$$

  where $d_{obs} = x_{\text{swFoot}} = 0\ [m]$ is the horizontal position of the ellipse obstacle, $w_{obs} = 0.2\ [m]$ and $h_{obs} = 0.04\ [m]$ are width and height of the ellipse.

- Swing foot vertical velocity must be greater or equal to zero:

$$\dot{x}_{\text{swFoot}} \geq 0 \tag{2.22}$$

- Vertical acceleration of CoM should be negative in the single stance phase so that system doesn't try to lift the CoM up when there is only one leg on the ground:

$$\ddot{y}_{\text{CoM},ss} \leq 0\ [m/s^2] \tag{2.23}$$

Boundary constraints:

- Swing foot starts on the ground from a stationary position in the beginning of the single stance phase and touches the ground at the end of the single stance phase:

$$\begin{aligned} y_{\text{swFoot}}(0) &= 0 \\ \dot{x}_{\text{swFoot}}(0) &= 0 \\ \dot{y}_{\text{swFoot}}(0) &= 0 \\ y_{\text{swFoot}}(T_1) &= 0 \end{aligned} \tag{2.24}$$

- Initial step length and the final step length should be same (for cyclic walking):

$$x_{\text{foot}} - x^-_{\text{foot}} = x^+_{\text{foot}} - x_{\text{foot}} \tag{2.25}$$

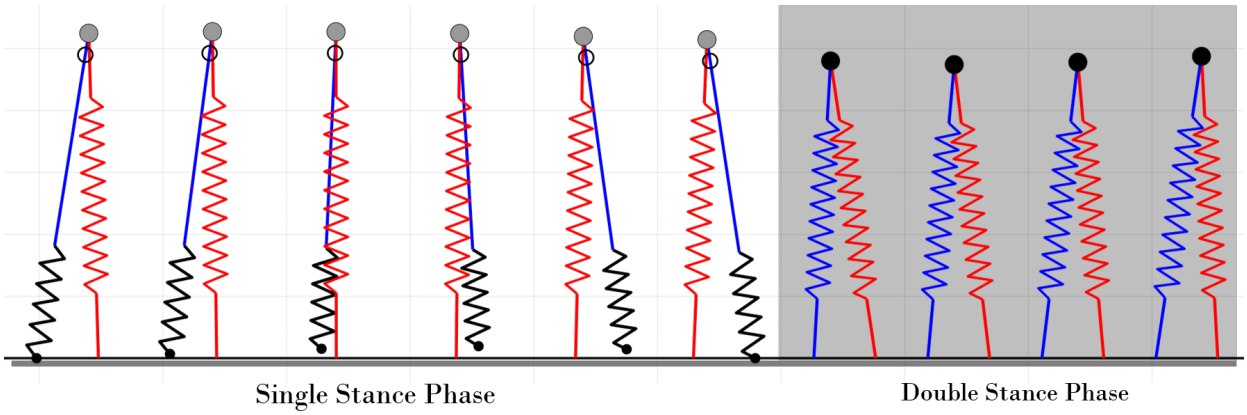


Figure 2.6: Snapshots of SLIP-SL's one step where gray dot in the single stance phase indicates the position of the point mass 'M' and the circle indicates the position of CoM

- The initial position of the main mass relative to the stance foot should be the same as the final one (for cyclic walking):

$$x_{stFoot,0} - x_M(0) = x_{stFoot}(T_2) - x_M(T_2) \tag{2.26}$$

- Constraints for cyclic walking:

$$\begin{aligned} y_{\text{CoM}}(0) &= y_{\text{CoM}}(T_2) \\ \dot{x}_{\text{CoM}}(0) &= \dot{x}_{\text{CoM}}(T_2) \\ \dot{y}_{\text{CoM}}(0) &= \dot{y}_{\text{CoM}}(T_2) \end{aligned} \tag{2.27}$$

- At the stage transition, CoM position and velocity were constrained to be continuous.
- At the end of the double stance phase, swing leg should be ready to lift off, i.e. swing leg spring should be at its free length:

$$L_{\text{sw,ds}}(T_2) = L_{0,\text{ds}} \tag{2.28}$$

Parameters to be optimized are spring stiffness values $k_{0,ss}$, $k_{0,ds}$, k_{swFoot}, k_{swLeg}, their respective free positions $L_{0,ss}$, $L_{0,ds}$, r_0,θ_0 and the initial conditions.

The optimization was conducted on MATLAB 2019b software by using 10 collocation points for each stage. Resulting spring parameters can be seen in Figure 2.7 for various trajectories and a snapshot of SLIP-SL's one step can be seen in Figure 2.6 for a sample trajectory. Trajectory 'A' from Figure 2.7 will be used as the reference in Section 2.4 where the step size was also constrained to 0.25 $[m]$ to avoid large ankle torques. The constant mechanical parameters of SLIP-SL are given in Table 2.1.

Table 2.1: SLIP-SL's constant mechanical parameters

m_M : 70 $[kg]$	m_{swLeg} : 7 $[kg]$
m_{swFoot} : 3 $[kg]$	L_{thigh} : 0.7 $[m]$
$I_{\text{swLeg}} = m_{\text{swLeg}} l_{\text{thigh}}^2/12$ $[kg \cdot m^2]$	$I_{\text{swFoot}} = m_{\text{swFoot}} L_{\text{thigh}}^2$

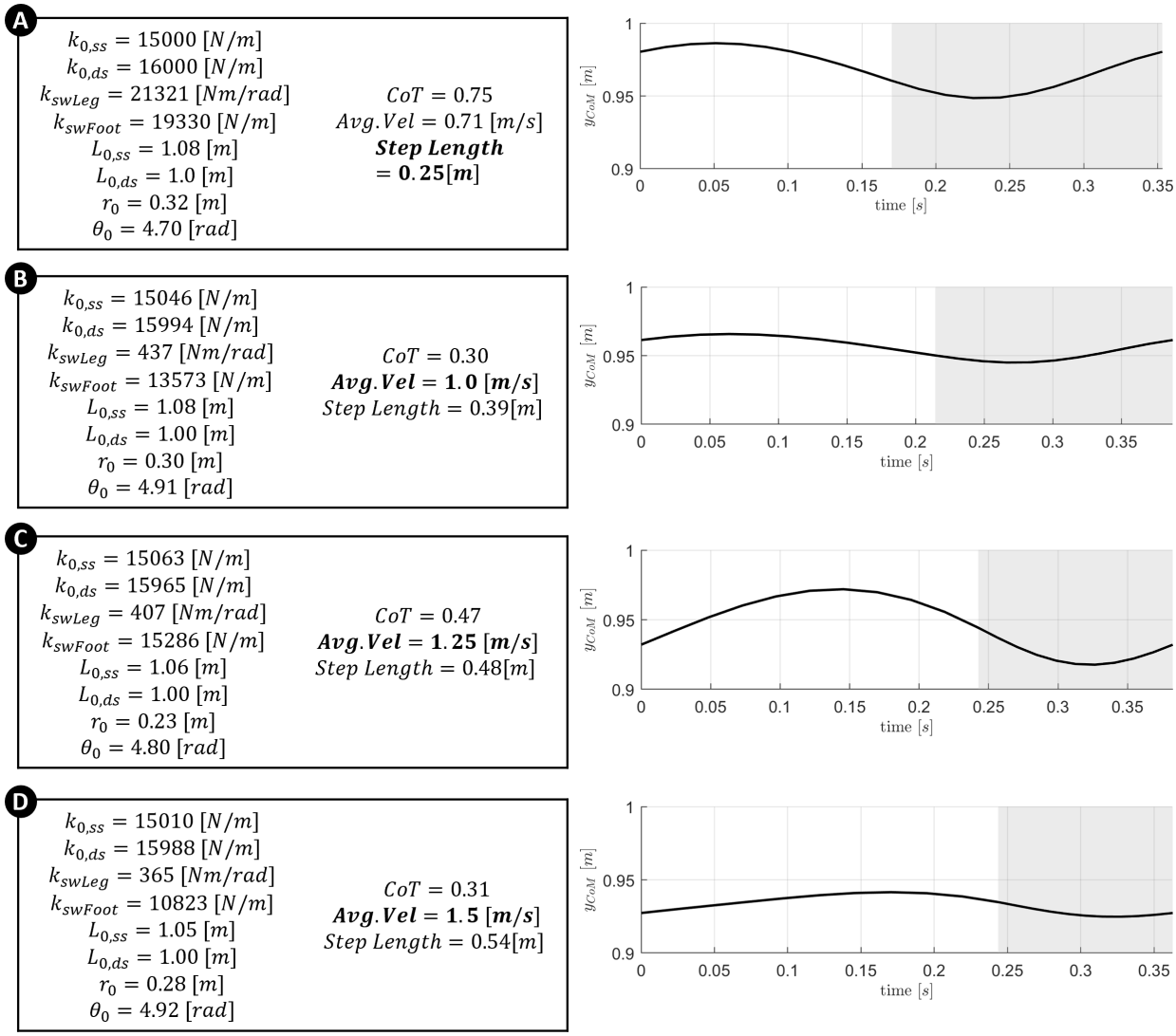


Figure 2.7: Optimization results for various SLIP-SL trajectories. For the trajectory A, step length was constrained to 0.25 [m] to get better ankle torques and this trajectory is the one that was used as reference in Section 2.3. For B, C and D trajectories, average velocity constraints were added. In this figure, resulting mechanical parameters, costs of transport and step length are given as well as the SLIP-SL's y_{CoM} trajectory. In the plots on the right, gray background means that the SLIP-SL is in double stance phase.

2.3 Feedback Linearization Control

In this section, the proposed controller will be introduced so that 5 linked bipedal robot model can track the reference SLIP-SL trajectories. The controller uses the feedback linearization notion, in a similar manner to [11] where a total energy control approach was used with the bipedal SLIP model. However, in this chapter, a trajectory tracking approach will be used.

2.3.1 Single Stance Phase

For the control of the robot in the single stance phase, there are three main tasks: tracking CoM trajectory $\boldsymbol{x}_G \in \mathbb{R}^2$, tracking swing foot trajectory $\boldsymbol{\xi} \in \mathbb{R}^2$, controlling the trunk orientation $\theta_5 \in \mathbb{R}$. The velocities related to these tasks can be calculated as:

$$\dot{\boldsymbol{x}}_{t,\text{ss}} = \boldsymbol{J}_{t,\text{ss}}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{2.29}$$

where $\dot{\boldsymbol{x}}_{t,\text{ss}} = [\dot{\boldsymbol{x}}_G, \dot{\boldsymbol{\xi}}, \dot{\theta}_5]^T$ is the velocity in the task space where subscript "ss" indicates the single stance phase and $\boldsymbol{J}_{t,ss}(\boldsymbol{q}) = [\boldsymbol{J}_G, \boldsymbol{J}_\xi, \boldsymbol{J}_{\theta_5}]^T$ are the combination of Jacobian matrices. $\boldsymbol{J}_G$ maps generalized velocities to the velocity of the center of mass, $\boldsymbol{J}_\xi$ maps generalized velocities to swing foot velocities and $\boldsymbol{J}_{\theta_5}$ maps generalized velocity to the trunk's angular velocity. By taking the time derivative of Equation (2.29) and inserting the obtained $\ddot{\boldsymbol{q}}$ into Equation (2.7) we can get:

$$\ddot{\boldsymbol{x}}_t = \boldsymbol{J}_{t,\text{ss}}\boldsymbol{M}^{-1}(\boldsymbol{S}\boldsymbol{u} - \boldsymbol{C}\dot{\boldsymbol{q}} - \boldsymbol{G}) + \dot{\boldsymbol{J}}_{t,\text{ss}}\dot{\boldsymbol{q}}. \tag{2.30}$$

Inputs should be chosen as:

$$\boldsymbol{u} = \boldsymbol{S}^{-1}\left(\boldsymbol{M}\boldsymbol{J}_{t,\text{ss}}^{-1}(\ddot{\boldsymbol{x}}_{t_d,ss} - \dot{\boldsymbol{J}}_{t,\text{ss}}\dot{\boldsymbol{q}}) + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{G}\right), \tag{2.31}$$

to get $\ddot{\boldsymbol{x}}_t = \ddot{\boldsymbol{x}}_{t_d,\text{ss}}$. By choosing:

$$\ddot{\boldsymbol{x}}_{t_d,\text{ss}} = \begin{bmatrix} K_{P_G}(x_{\text{CoM,des}} - x_{\text{CoM}}) + K_{D_G}(\dot{x}_{\text{CoM,des}} - \dot{x}_{\text{CoM}}) \\ K_{P_G}(y_{\text{CoM,des}} - y_{\text{CoM}}) + K_{D_G}(\dot{y}_{\text{CoM,des}} - \dot{y}_{\text{CoM}}) \\ K_{P_{\text{sw}}}(x_{\text{sw,des}} - x_{\text{sw}}) + K_{D_{\text{sw}}}(\dot{x}_{\text{sw,des}} - \dot{x}_{\text{sw}}) \\ K_{P_{\text{sw}}}(y_{\text{sw,des}} - y_{\text{sw}}) + K_{D_{\text{sw}}}(\dot{y}_{\text{sw,des}} - \dot{y}_{\text{sw}}) \\ K_{P_T}(\theta_{5,\text{des}} - \theta_5) + K_{D_T}(-\dot{\theta}_5) \end{bmatrix} \tag{2.32}$$

where K_P and K_D are the proportional and derivative gains for the controller and $\theta_{5,des}$ is the desired trunk angle, desired trajectories can be tracked. Desired trajectories will be chosen as the SLIP-SL trajectories that were obtained in Section 2.2. In this study, the desired trunk angle was chosen as $\theta_{5,\text{des}} = \pi$ [rad].

2.3.2 Double Stance Phase

During the double stance phase, swing foot remains on the ground which means there is one less task to be carried out. This means that some modifications should be made to the single stance phase controller so that it can be used in the double stance phase. The task space for the double stance phase then becomes:

$$\dot{\boldsymbol{x}}_{t,ds} = \begin{bmatrix} \dot{\boldsymbol{x}}_G \\ \dot{\theta}_5 \end{bmatrix} \in \mathbb{R}^3. \tag{2.33}$$

This reduction in dimension of the task space is somewhat problematic since we need to determine 5 separate inputs but the dimension of $\dot{\boldsymbol{x}}_{t,ds}$ is 3 which means there is more than one correct way to allocate the inputs. To calculate the proper inputs, the following equation can be used:

$$\dot{\boldsymbol{q}}_{\text{hip}} = \boldsymbol{J}_{\text{hip,sw}}(\boldsymbol{q}_{\text{sw}})\dot{\boldsymbol{q}}_{\text{sw}} = \boldsymbol{J}_{\text{hip,st}}(\boldsymbol{q}_{\text{st}})\dot{\boldsymbol{q}}_{\text{st}}, \tag{2.34}$$

where $\boldsymbol{J}_{\text{hip, }i}(\boldsymbol{q}_i)$ is the jacobian matrix that maps the corresponding legs angular velocities, $\dot{\boldsymbol{q}}_{\text{st}} = [\dot{\theta}_1, \dot{\theta}_2]^T$ and $\dot{\boldsymbol{q}}_{\text{sw}} = [\dot{\theta}_3, \dot{\theta}_4]^T$, to the velocity of the hip. This holds since both foot are on the ground during the double stance phase. From equation (2.34),

$$\dot{\boldsymbol{q}}_{\text{st}} = \boldsymbol{J}^{-1}_{\text{hip,st}}(\boldsymbol{q}_{\text{st}})\boldsymbol{J}_{\text{hip,sw}}(\boldsymbol{q}_{\text{sw}})\dot{\boldsymbol{q}}_{\text{sw}}, \tag{2.35}$$

can be obtained to get:

$$\dot{\boldsymbol{q}} = \underbrace{\begin{bmatrix} \boldsymbol{J}^{-1}_{\text{hip,st}}\boldsymbol{J}_{\text{hip,sw}} & 0_{2\times1} \\ I_{2\times2} & 0_{2\times1} \\ 0_{1\times2} & 1 \end{bmatrix}}_{\boldsymbol{\Gamma}(\boldsymbol{q})} \underbrace{\begin{bmatrix} \dot{\boldsymbol{q}}_{\text{sw}} \\ \dot{\theta}_5 \end{bmatrix}}_{\dot{\boldsymbol{q}}_a}. \tag{2.36}$$

After taking the time derivative of (2.36), then substituting the $\ddot{\boldsymbol{q}}$ and $\dot{\boldsymbol{q}}$ terms from (2.36) into (2.7) and multiplying with $\boldsymbol{\Gamma}^T$ from left, following can be obtained:

$$\boldsymbol{M}_a(\boldsymbol{q})\ddot{\boldsymbol{q}}_a + \boldsymbol{C}_a(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}_a + \boldsymbol{G}_a(\boldsymbol{q}) = \boldsymbol{u}_a, \tag{2.37}$$

where

$$\begin{cases} \boldsymbol{M}_a = \boldsymbol{\Gamma}^T\boldsymbol{M}\boldsymbol{\Gamma} \\ \boldsymbol{C}_a = \boldsymbol{\Gamma}^T(\boldsymbol{C}\boldsymbol{\Gamma} + \boldsymbol{M}\dot{\boldsymbol{\Gamma}}) \\ \boldsymbol{G}_a = \boldsymbol{\Gamma}^T\boldsymbol{G} \\ \boldsymbol{u}_a = \boldsymbol{\Gamma}^T\boldsymbol{S}\boldsymbol{u} \end{cases} \tag{2.38}$$

Using a similar technique that was used for deriving (2.31):

$$\boldsymbol{u}_a = \boldsymbol{M}_a\boldsymbol{J}^{-1}_{t,\text{ds}}(\ddot{\boldsymbol{x}}_{t_d,\text{ds}} - \dot{\boldsymbol{J}}_{t,\text{ds}}\dot{\boldsymbol{q}}_a) + \boldsymbol{C}_a\dot{\boldsymbol{q}}_a + \boldsymbol{G}_a, \tag{2.39}$$

can be found where $\ddot{\boldsymbol{x}}_{t_d,\text{ds}}$ should be chosen as:

$$\ddot{\boldsymbol{x}}_{t_d,\text{ds}} = \begin{bmatrix} K_{P_G}(x_{\text{CoM, des}} - x_{\text{CoM}}) + K_{D_G}(\dot{x}_{\text{CoM}} - \dot{x}_{\text{CoM}}) \\ K_{P_G}(y_{\text{CoM, des}} - y_{\text{CoM}}) + K_{D_G}(\dot{y}_{\text{CoM, slipsl}} - \dot{y}_{\text{CoM}}) \\ K_{P_T}(\theta_{5,\text{des}} - \theta_5) + K_{D_T}(-\dot{\theta}_5) \end{bmatrix}. \tag{2.40}$$

However, only $\boldsymbol{u}_a \in \mathbb{R}^3$ can be obtained in this way which only has dimension 3. What we need to control the robot is $\boldsymbol{u} \in \mathbb{R}^5$. One way to calculate $\boldsymbol{u}$ is by using the relation $\boldsymbol{u}_a = \boldsymbol{\Gamma}^T\boldsymbol{S}\boldsymbol{u}$ given in (2.38) as:

$$\boldsymbol{u} = \boldsymbol{S}^{-1} \left(\underbrace{\boldsymbol{W}^{-1}\boldsymbol{\Gamma}(\boldsymbol{\Gamma}^T\boldsymbol{W}^{-1}\boldsymbol{\Gamma})^{-1}}_{(\boldsymbol{\Gamma}^T)^{+\boldsymbol{W}}} \boldsymbol{u}_a \right), \tag{2.41}$$

where $(\boldsymbol{\Gamma}^T)^{+\boldsymbol{W}}$ is the weighted matrix inverse operation. $\boldsymbol{W} \in \mathbb{R}^{5\times5}$ matrix can be used to penalize high input torques such as the ankle torque but we selected it as identity matrix for this chapter.

Controllers for the single and double stance phases are thus derived. Another important aspect of the controller in tracking the SLIP-SL trajectories is the switching of phases at the correct moments. When the biped robot is in the double stance phase and the tracked trajectory goes into single stance phase, controller switches to the single stance phase controller and commands the robot to lift its foot so it too can switch to the proper phase.

2.4 Simulation Results and Discussion

we will assess the performance of the designed controller in accurately tracking the reference SLIP-SL trajectories and evaluate the suitability of the SLIP-SL model as a template for walking. The mechanical parameters of the 5-link robot utilized in the study are presented in Table 2.3, while the chosen gain values for the controller are listed in Table 2.2. The torque limits on the motors are set at $200; [Nm]$ to ensure safe operation.

To determine the optimal gain values for the controller, we employed a particle swarm optimization (PSO) algorithm [12]. While stable gaits were initially attainable through manual tuning, the utilization of the PSO algorithm allowed for the achievement of improved walking efficiency by obtaining more refined controller parameters.

Through this simulation study, our objective is to validate the effectiveness of the designed controller in accurately tracking the desired SLIP-SL trajectories. Additionally, we aim to demonstrate the suitability of the SLIP-SL model as a viable template for achieving stable and efficient walking behaviors within the 5-link robot system.

The simulations were implemented in MATLAB 2020b's Simulink environment with ode45 solver and variable step settings (absolute tolerance was set to 1e-8).

Figure 2.8 shows the resulting CoM trajectory and trunk orientation the controlled system

Table 2.2: Control Parameters

$K_{P_G} : 54$	$K_{D_G} : 9$
$K_{P_{\text{sw}}} : 82$	$K_{D_{\text{sw}}} : 8$
$K_{P_T} : 36$	$K_{D_T} : 4$

Table 2.3: 5 Link Model Parameters

$l_1 = l_4 : 0.48\ [m]$	$l_2 = l_3 : 0.48\ [m]$	$l_5 : 0.48\ [m]$
$m_1 = m_4 : 5\ [kg]$	$m_2 = m_3 : 5\ [kg]$	$m_5 : 60\ [kg]$
$I_i = m_i l_i^2/12\ [kg \cdot m^2],\ i = 1, 2, 3, 4, 5$		

when the proposed controller is used. Figure 2.9 shows the swing foot trajectories and Figure 2.10 shows the input torques for the same system. It can be seen that the proposed controller does a good job in tracking the reference trajectories of the SLIP-SL template, which were obtained in Section 2.2. The reference SLIP-SL trajectories for the swing foot would not be available if a template such as the popular SLIP was used.

It can be seen in Table 2.2 that relatively low gains were chosen for this study. Tracking performance can be increased by using larger gains but since this model is fully actuated and has an ankle torque, zero moment point (ZMP) condition must also be checked. ZMP criterion states that if the center of pressure moves to the toe (or to the “outside" of the foot), foot would rotate and system would be under actuated [13]. For this trajectory, center of pressure stays within a 30 [cm] foot.

Figure 2.11 shows snapshots of one step of the 5 link models gait. to check the stability of the gait, Poincaré map approach was considered [11]. The dimensions of the Poincaré map were selected as the $\theta_P = \text{atan2}(\dot{y}_{\text{CoM}}, \dot{x}_{\text{CoM}})$, y_{CoM} and total energy of the 5 link model E at the vertical leg orientation (VLO). VLO happens when CoM of the 5 link model is at the same horizontal position as the stance foot. VLO was chosen as the Poincaré section because the horizontal position doesn't need to be considered at this point. Poincaré stability criterion indicates that if the return map converges to a fixed point, a hybrid system with impact effects can be considered periodic [11]. Poincare Map for the controlled 5 link model is shown in Figure 2.12. It can be seen that the gait converges to a stable point in the section after a couple of steps which indicates stability.

The CoT value for the reference SLIP-SL trajectory was 0.7520 and this value is 0.7745 for the controlled 5 link robot. Also, the average velocity of the SLIP-SL trajectory was 0.7080 [m/s] and this value was 0.6974 [m/s] for the controlled system. These values being very similar between the reference and the controlled model also indicates the validity of the proposed controller. Cost of transport being slightly higher is expected because the 5 linked model needs to keep its body upright but SLIP-SL doesn't have this issue.

In this chapter, it was also shown that by using direct collocation optimization, various SLIP-SL trajectories can be obtained (Figure 2.7) that resemble the walking gait. Cost of transport tended to decrease when the average velocity of the gait was increased and step length was not constrained but this kind of trajectories can be more demanding on the inputs and they sometimes resulted in large ankle torques which was troubling for the ZMP criterion. Also stiffness of the legs were limited to $15000\ [Nm] \leq k_{0,i} \leq 16000\ [Nm],\ i \in \{ds, ss\}$ to keep the CoM height within a certain range.

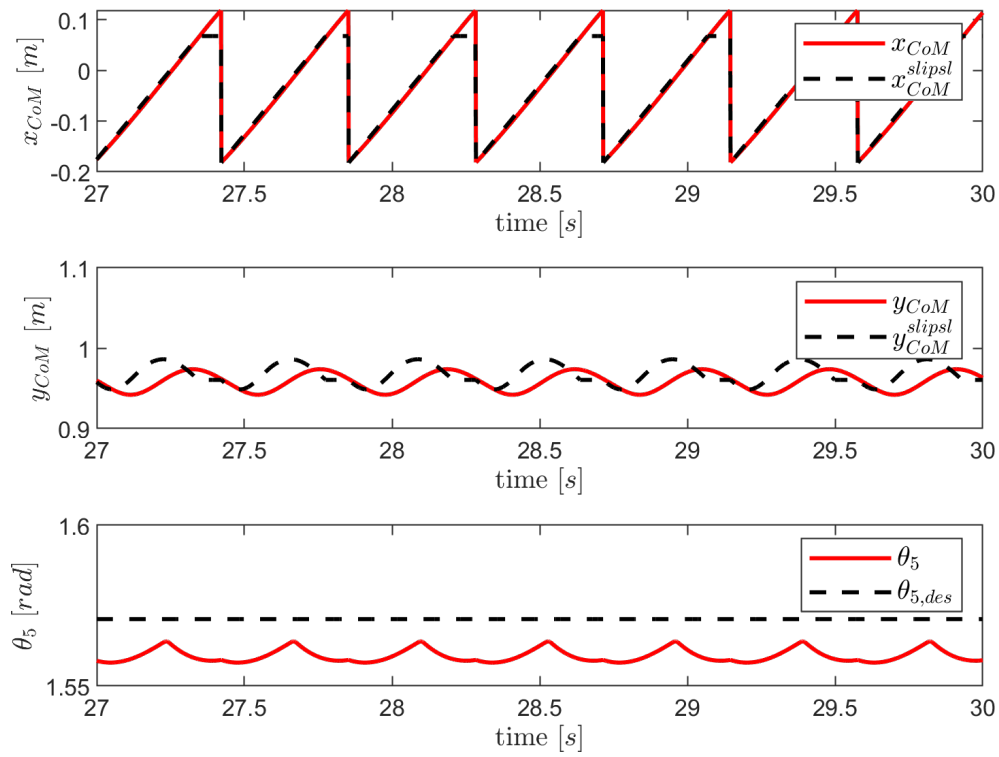


Figure 2.8: Trajectory tracking results for CoM horizontal position, vertical position and trunk orientation

This stiffness limits was chosen to have a similar height trends with [11] and [5] where k_0 was 15696 [Nm]. It can be seen that $k_{0,ss} \neq k_{0,ds}$ for the given trajectories but it was possible to find feasible trajectories with $k_{0,ss} = k_{0,ds}$, even with $k_{0,ss} = k_{0,ds} = 15696\ [Nm]$ but this doesn't really mean that linear leg springs were the same in single and double stance phase since $L_{0,ss} \neq L_{0,ds}$. The free length of the springs in the double stance phase was set to 1 [m] to be the same with [11] but $L_{0,ss}$ needs to be larger than this value to keep the CoM high enough. If $L_{0,ss}$ was 1.0 [m], CoM would sag further than desired range in the single stance phase.

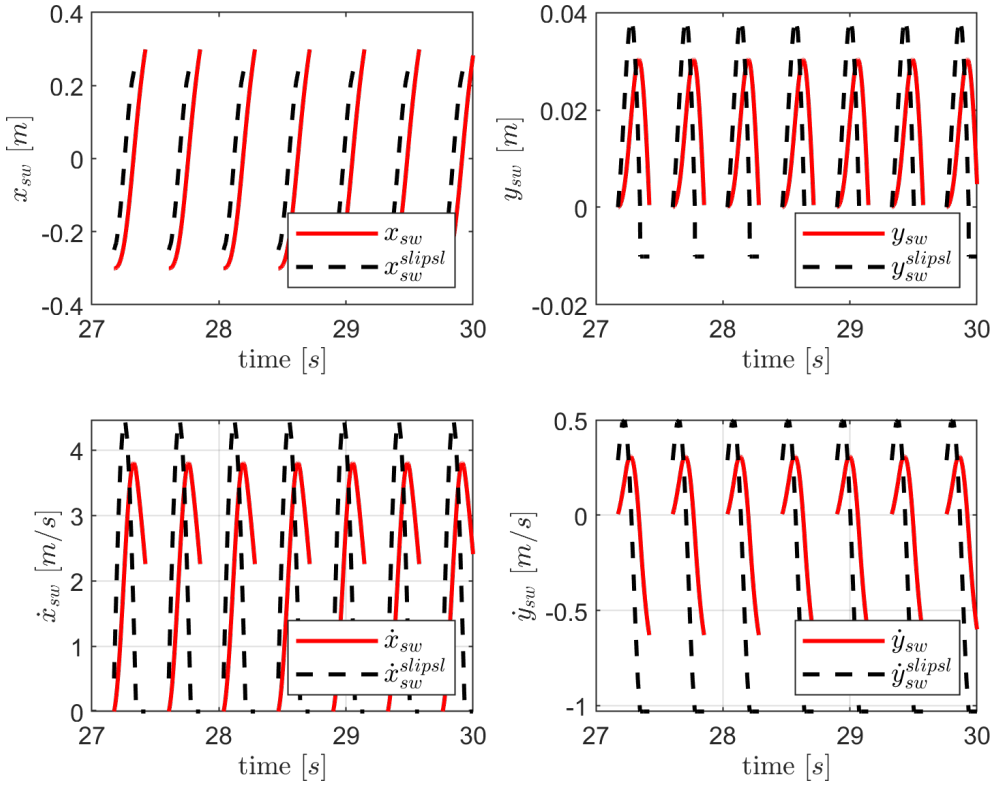


Figure 2.9: Trajectory tracking results for the swing foot

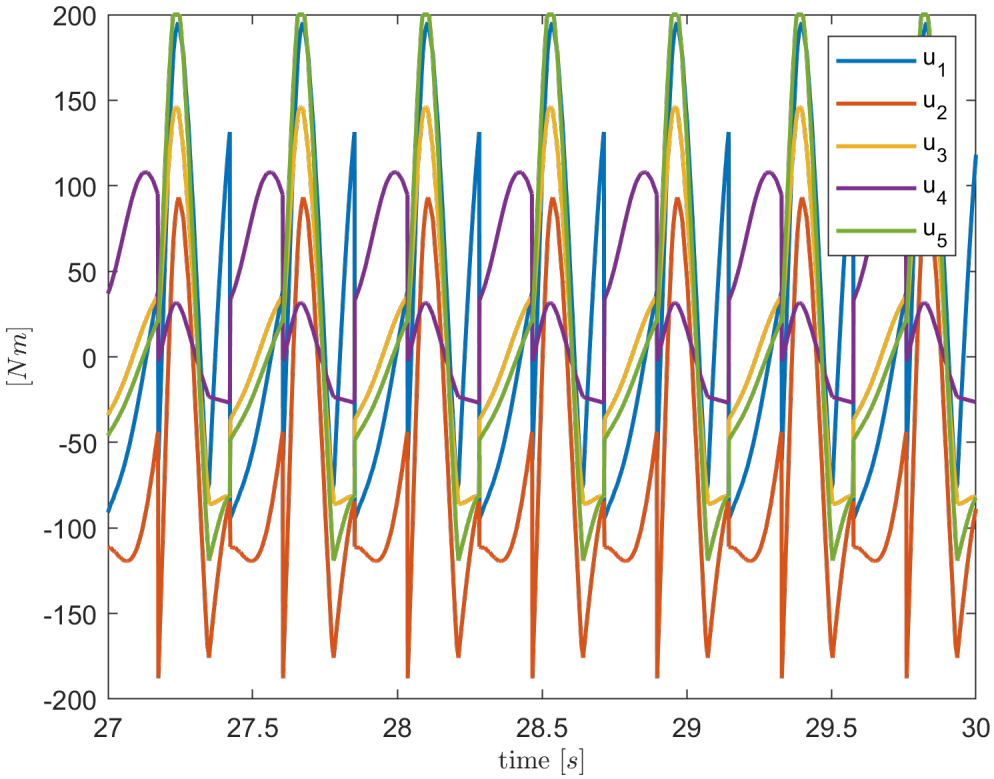


Figure 2.10: Motor torques

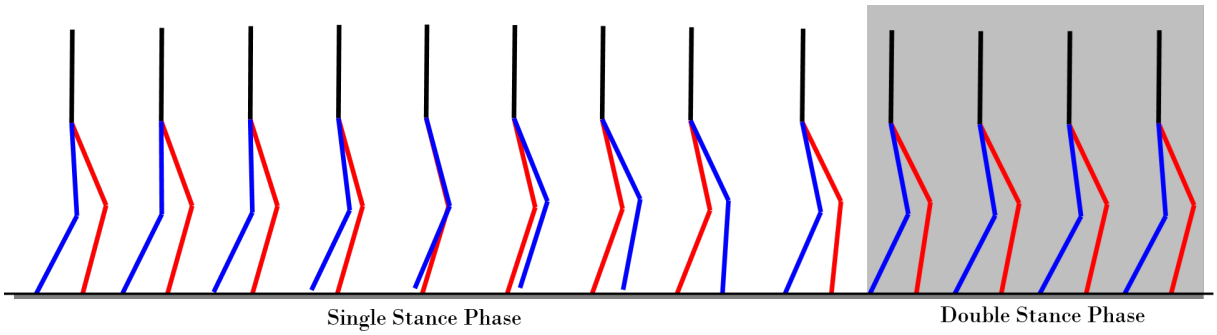


Figure 2.11: Snapshots from a step of the 5 Link model

2.5 Conclusion

In this chapter, we have introduced a novel template model called SLIP-SL, which serves as an extension to the well-established Spring-Loaded Inverted Pendulum (SLIP) model. While the SLIP model is capable of generating reference trajectories for the center of mass (CoM) that can mimic the two-phased walking observed in animals and humans, it lacks the dynamics of the swing leg. Therefore, when utilizing the SLIP model as a template, additional steps must be taken to obtain the reference trajectory for the swing foot, enabling control of an actual robot.

Recognizing the importance of accounting for the swinging motion as a crucial aspect of the walking gait, we have proposed the SLIP-SL model. This extension incorporates the dynamics of the swing leg while maintaining the passivity of the original SLIP model. By introducing swing leg dynamics into the SLIP-SL model, we provide a more comprehensive representation of the walking process, allowing for a more accurate control of the actual robot.

In order to achieve stable and cyclic walking with the passive SLIP-SL model, it is crucial to determine the appropriate model parameters and initial conditions. We've demonstrated that direct collocation methods are an effective approach for finding these parameters. This step is significant due to the numerous parameters involved, and other exhaustive search methods may not yield feasible results or convergence.

The application of direct collocation methods allows for the generation of diverse trajectories with varying average velocities and center of mass (CoM) behaviors, all based on the same underlying principles. This flexibility enables the exploration of different walking patterns and behaviors for the SLIP-SL model.

To track the obtained reference trajectories we proposed a feedback linearization based controller. Our simulation experiments shows that this controller can successfully guide a 5-link biped robot to achieve a stable gait while adhering to the Zero Moment Point (ZMP) criterion. These results further confirmed the suitability of the SLIP-SL model as a template for walking robots.

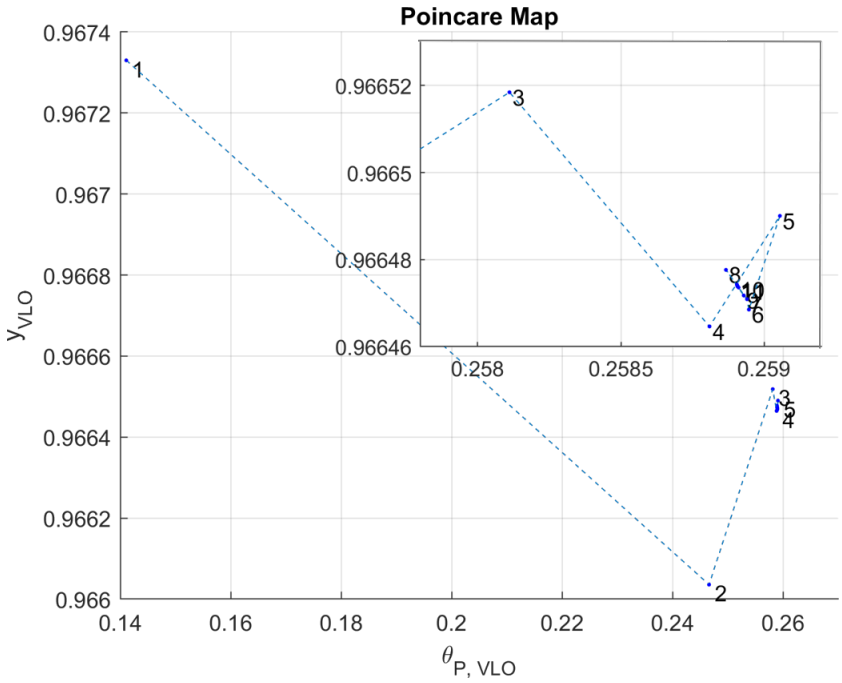


Figure 2.12: 2D section of the Poincaré Map where the numbers indicate the step number (zoomed in version is shown in the right upper corner of the figure)

Overall, this research showcases the feasibility of using the SLIP-SL model to achieve stable walking in bipedal robots, while also providing insights into the versatility and adaptability of the SLIP-SL model for various walking patterns and behaviors.

Chapter 3

Variable Stiffness Spring Loaded Inverted Pendulum Model with Swing Leg Dynamics (VSLIP-SL)

In Chapter 2, we've shown the effectiveness of the proposed SLIP-SL model. It was able to simultaneously generate the center of mass (CoM) and swing leg trajectories that are necessary components for controlling a bipedal robot model. This extended model is useful but it still lack robustness against disturbances.

The ability of humans to adjust their leg stiffness to effectively mitigate disturbances has been well-documented [14]. In an effort to capture this aspect of human locomotion, researchers proposed an extension to the SLIP model known as Variable Leg Stiffness SLIP (V-SLIP) in studies such as [15] and [16]. By incorporating variable leg stiffness into the SLIP model, the V-SLIP model aimed to emulate the human capability to adapt leg stiffness in response to disturbances. The proposed controller for the V-SLIP model demonstrated its efficacy in successfully handling disturbances through the utilization of variable stiffness responses.

However, it is important to acknowledge that the V-SLIP model, similar to the SLIP model, does not consider the dynamics of the swing foot, which is a critical element in bipedal gaits. The swing foot dynamics play a significant role in achieving stable and natural walking patterns. Thus, the inclusion of swing foot dynamics is crucial for developing more comprehensive models of bipedal locomotion.

The objective of this chapter is twofold: firstly, to expand the V-SLIP model by incorporating swing leg dynamics, resulting in the development of a new model referred to as VSLIP-SL (in other words, extend the SLIP-SL model to have variable stiffness). Secondly, we will propose a controller for the higher-order bipedal robot model based on this extended template. The bipedal robot model under consideration is a 5-link fully actuated system devoid of compliant elements.

The primary challenge for the controller lies in translating the variable stiffness responses

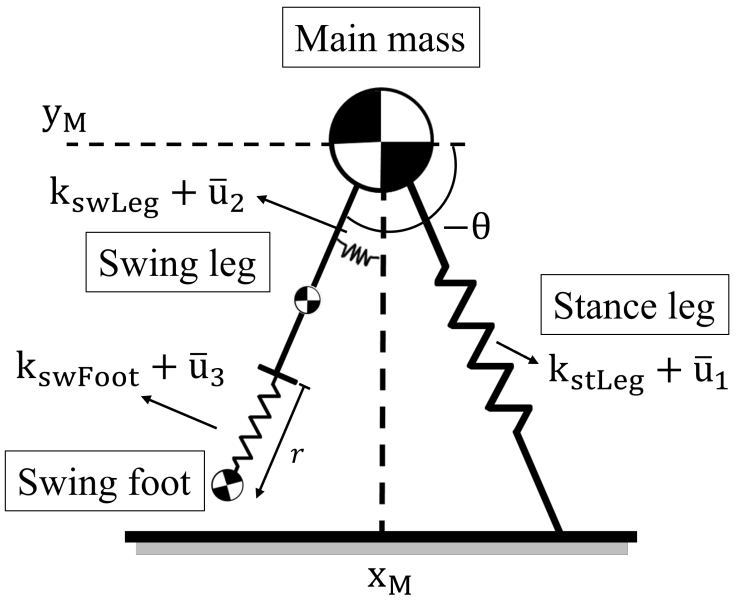


Figure 3.1: The main goal of this chapter is to reshape the dynamics of 5-link bipedal robot model (on the right) so that it is as close as possible to the variable stiffness spring-loaded inverted pendulum model with swing legs (VSLIP-SL, on the left)

obtained from the VSLIP-SL model into appropriate actuation torques for the bipedal robot. To address this challenge, we propose a feedback linearization controller augmented with an encoding-decoding scheme. This controller is intended to facilitate the mapping of the variable stiffness characteristics of the VSLIP-SL model into effective torque inputs for the actuators of the bipedal robot.

Through the integration of swing leg dynamics into the V-SLIP model and the design of the proposed feedback linearization controller, we aim to enhance the capabilities of bipedal robots in terms of achieving stable and robust walking patterns.

This chapter is organized as follows: Section 3.1 introduces the VSLIP-SL and 5-link bipedal robot models, Section 3.2 introduces the proposed controllers and Section 3.3 presents the simulation results.

3.1 Systems and Modeling

In this section, we will describe the variable stiffness spring loaded inverted pendulum model with swing leg dynamics (VSLIP-SL) and the 5-link bipedal robot model and their dynamics.

3.1.1 VSLIP-SL Model

Like the human gait, VSLIP-SL's walking consists of two phases: a single stance phase (SS) where only one foot is in contact with the ground and a double stance phase (DS) where

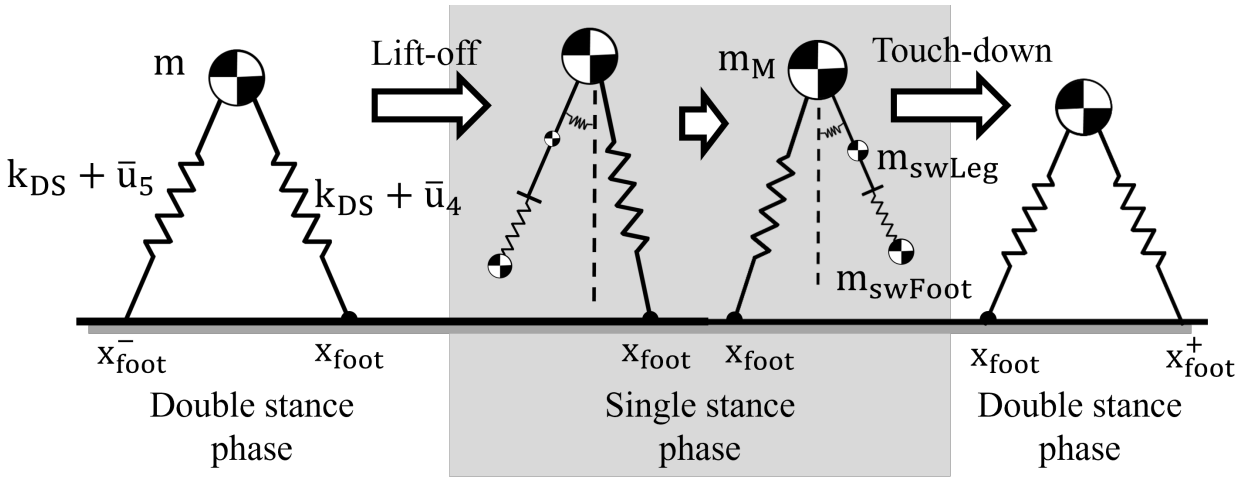


Figure 3.2: VSLIP-SL model in the double stance phase and the single stance phase

both legs are in contact with the ground.

The model is shown in the single stance phase configuration in Figure 3.1. In this phase, it consists of three massed elements: a main point mass, a massed swing leg and another point mass representing the swing foot. This model moves with the help of three springs. The stance leg linear spring is connected to the ground and the vertical axis that passes through the main mass, the torsional swing leg spring is connected between the main mass and the swing leg, and the linear swing foot spring is connected between the swing leg and the swing foot.

VSLIP-SL in the double stance phase is shown in Fig 3.2. Transition from single stance phase to double stance phase happens when the swing foot touches the ground. This event is called "the touch-down event". The position and the velocity of the center of mass (CoM) is conserved during this transition. The model in the double stance phase is equivalent to the traditional SLIP model. It consists of a single point mass representing the CoM and two linear springs representing the legs. When the spring in the back reaches its free length, lift-off event happens and the model goes back to the single stance phase. We assume that we can control the stiffnesses of all the springs through the inputs $\bar{\boldsymbol{u}}_{\text{ss}}$ and $\bar{\boldsymbol{u}}_{\text{ds}}$.

The dynamic model for the single support phase can be written as:

$$\overline{\boldsymbol{M}}(\bar{\boldsymbol{q}})\ddot{\bar{\boldsymbol{q}}} + \overline{\boldsymbol{H}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) = \bar{\boldsymbol{S}}\bar{\boldsymbol{u}}_{\text{ss}}, \tag{3.1}$$

where $\bar{\boldsymbol{q}} = [x_M, y_M, \theta, r]^T$ are the generalized coordinates, $\overline{\boldsymbol{M}}(\bar{\boldsymbol{q}}) \in \mathbb{R}^{4\times4}$ is an inertia matrix, $\overline{\boldsymbol{H}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \in \mathbb{R}^4$ is a Coriolis, centrifugal and gravity terms vector, $\bar{\boldsymbol{u}}_{\text{ss}} = [\bar{u}_1, \bar{u}_2, \bar{u}_3]^T \in \mathbb{R}^3$ are the variable stiffness inputs that we have introduced for this study and $\bar{\boldsymbol{S}} \in \mathbb{R}^{4\times3}$ is the appropriate mapping matrix for them.

The dynamic model for the double stance phase can be written as:

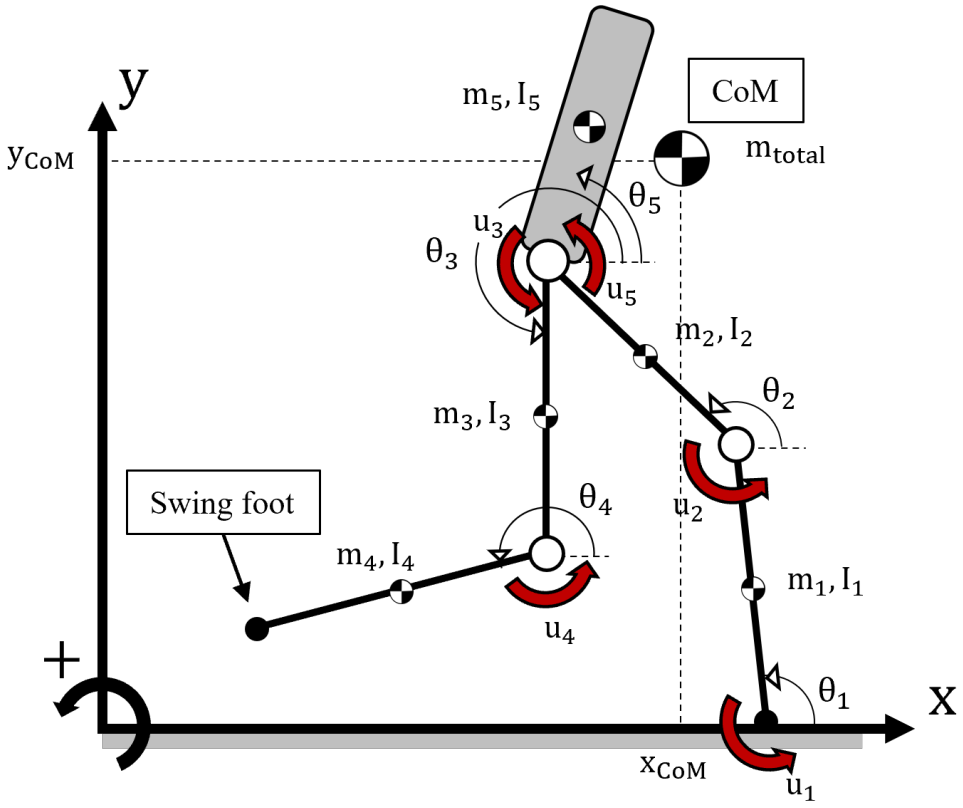


Figure 3.3: 5-link fully actuated model. The inputs are indicated by the red arrows.

$$m \begin{bmatrix} \ddot{\bar{x}}_{\text{CoM}} \\ \ddot{\bar{y}}_{\text{CoM}} \end{bmatrix} - \boldsymbol{F}_{\text{sw}} - \boldsymbol{F}_{\text{st}} - m\boldsymbol{g} = \boldsymbol{B}\bar{\boldsymbol{u}}_{\text{ds}}, \tag{3.2}$$

where $m = m_{\text{M}} + m_{\text{swLeg}} + m_{\text{swFoot}}$ is the total mass of the system, $\bar{x}_{\text{CoM}}$ and $\bar{y}_{\text{CoM}}$ are the horizontal and vertical positions of the center of mass, $\boldsymbol{g} = [0, -9.81]^T$ [m/s^2] is the gravitational acceleration, $\boldsymbol{F}_{\text{st}}$ is the force generated by the nominal stance leg spring and $\boldsymbol{F}_{\text{sw}}$ is the force generated by the nominal swing leg spring. $\bar{\boldsymbol{u}}_{\text{ds}} = [\bar{u}_4, \bar{u}_5]^T$ are the variable stiffness terms for the double stance phase and $\boldsymbol{B} \in \mathbb{R}^{2\times 2}$ is their mapping matrix.

A more detailed explanation for the SLIP-SL model (without the variable stiffness) can be found in [17].

3.1.2 5-Link Bipedal Robot Model

The 5-link bipedal robot model used in this chapter can be seen in Figure 3.3. It has 5 links which are connected with revolute joints. The model is constricted to the sagittal plane. It has 5 actuators: 2 at the hip, 2 at the knees and 1 at the "ankle", meaning the system is fully actuated. It is also a hybrid dynamical system; it switches between the single support phase and the double support phase through the discrete touch-down and lift-off events. This model is similar to the one used in [11] and in Chapter 2.

The dynamic model of the 5-link model in the single stance phase can be written as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{H}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{S}\boldsymbol{u} \tag{3.3}$$

where $\boldsymbol{q} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^T \in \mathbb{R}^5$ are the generalized coordinates, $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{5\times5}$ is the inertia matrix, $\boldsymbol{H}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^5$ is the Coriolis, centrifugal and gravitational terms vector, $\boldsymbol{S} \in \mathbb{R}^{5\times5}$ is the distribution matrix of the inputs and $\boldsymbol{u} \in \mathbb{R}^5$ are the input torques. The description of the generalized coordinates and input torques can be seen in Figure 3.3.

When the swing foot touches the ground, a discrete collision event occurs and the model goes to the double support phase. The generalized momentum of the model changes discontinuously. We model this by an impulsive impact force as:

$$\boldsymbol{M}(\boldsymbol{q})\Delta\dot{\boldsymbol{q}} = \boldsymbol{J}_c^T \boldsymbol{\lambda}_{\text{impact}}, \tag{3.4}$$

where $\boldsymbol{J}_c \in \mathbb{R}^{2\times5}$ is a constraint Jacobian matrix that maps the joint velocities to the swing foot velocity in horizontal and vertical directions. The generalized reaction forces in x and y directions are indicated as $\boldsymbol{\lambda}_{\text{impact}} = [\lambda^x_{\text{impact}}, \lambda^y_{\text{impact}}]^T \in \mathbb{R}^2$. Assuming that the impact is inelastic, velocity of the swing foot touching the ground will become zero after the impact which can be written as:

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}^+ = 0 \Leftrightarrow \boldsymbol{J}_c(\boldsymbol{q})\Delta\dot{\boldsymbol{q}} = -\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}^-, \tag{3.5}$$

where "$-$" indicates the moment just before the impact and "$+$" just after the impact. By solving (3.4) and (3.5) for impact force, we can obtain:

$$\boldsymbol{\lambda}_{\text{impact}} = -(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}\boldsymbol{J}_c\dot{\boldsymbol{q}}^-. \tag{3.6}$$

By inserting $\boldsymbol{\lambda}_{\text{impact}}$ into (3.4), the generalized velocities just after the impact can be found as:

$$\dot{\boldsymbol{q}}^+ = (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{J}_c^T(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}\boldsymbol{J}_c)\dot{\boldsymbol{q}}^-, \tag{3.7}$$

At the moment of impact, generalized velocities change discontinuously while position of the model remains the same. However, we switch the definitions of the legs; after the touch-down, the former swing leg becomes the stance leg and vice versa. This change in the generalized coordinates can be formulated as:

$$\boldsymbol{x}^+ = \boldsymbol{f}_H(\boldsymbol{x}^-) \tag{3.8}$$

where $\boldsymbol{x} = [\boldsymbol{q}^T, \dot{\boldsymbol{q}}^T]^T$.

To model the double stance phase, a constraint force $\boldsymbol{\lambda}_{\text{ds}} = [\lambda^x_{\text{ds}}, \lambda^y_{\text{ds}}]^T \in \mathbb{R}^2$ is introduced to keep the swing foot on the ground. With the non-slip assumption, double stance phase dynamics can be written with the help of the constraint:

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0. \tag{3.9}$$

The dynamical model for the double stance phase can be written as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{H}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{S}\boldsymbol{u} + \boldsymbol{J}_c^T \boldsymbol{\lambda}_{\mathrm{ds}}. \tag{3.10}$$

The constraint force $\boldsymbol{\lambda}_{\mathrm{ds}}$ can be obtained by taking the time derivative of equation (3.9) and inserting it into (3.10) as follows:

$$\boldsymbol{\lambda}_{\mathrm{ds}} = -(\boldsymbol{J}_c \boldsymbol{M}^{-1} \boldsymbol{J}_c^T)^{-1} (\boldsymbol{J}_c \boldsymbol{M}^{-1} (\boldsymbol{S}\boldsymbol{u} - \boldsymbol{H}) + \dot{\boldsymbol{J}}_c \dot{\boldsymbol{q}}). \tag{3.11}$$

One walking cycle is completed when the lift-off event happens at the end of the double stance phase. This event occurs when the $\lambda_{ds}^y > 0$ condition is satisfied and the system goes back to the single stance phase.

3.2 Control

3.2.1 VSLIP-SL Controller

Two different controllers will be designed for the two phases but the general idea remains the same. In [15], a controller for the variable stiffness SLIP model was introduced. Controller proposed in this section is an extension of this to our VSLIP-SL model.

For the single stance phase, let's define the state as $\boldsymbol{z}_{\mathrm{ss}} = [\bar{\boldsymbol{q}}^T, \dot{\bar{\boldsymbol{q}}}^T]^T$. Then, we can write the dynamics of the single stance phase given by equation (3.1) in the state space form as:

$$\dot{\boldsymbol{z}}_{\mathrm{ss}} = \boldsymbol{f}_{\mathrm{ss}}(\boldsymbol{z}_{\mathrm{ss}}) + \boldsymbol{g}_{\mathrm{ss}}(\boldsymbol{z}_{\mathrm{ss}}) \bar{\boldsymbol{u}}_{\mathrm{ss}}, \tag{3.12}$$

where $\boldsymbol{g}_{\mathrm{ss}} = [\boldsymbol{g}_1, \boldsymbol{g}_2, \boldsymbol{g}_3] \in \mathbb{R}^{8\times 3}$.

In the single stance phase, there are three springs and their stiffness can be controlled. We want to control the vertical position of the main mass and the horizontal and vertical positions of the swing foot. To achieve this, outputs should be chosen as:

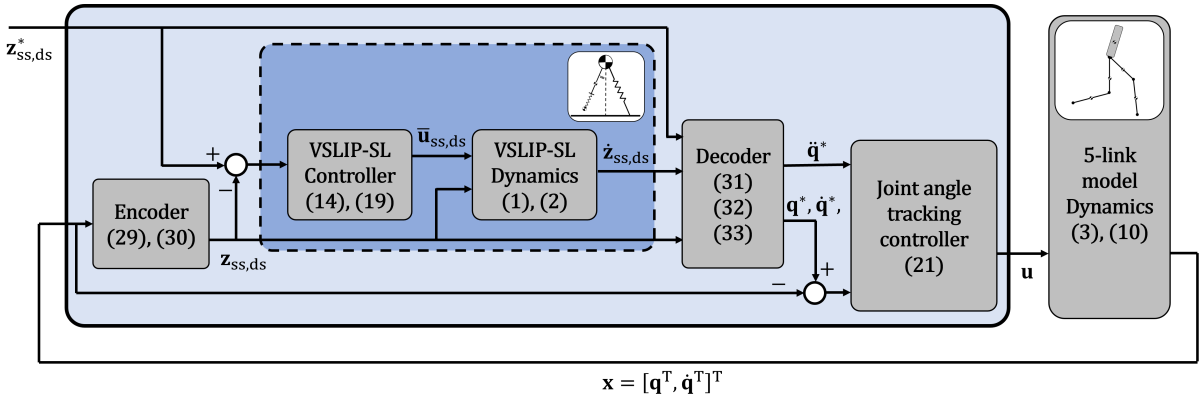


Figure 3.4: Diagram of the 5-link bipedal robot controller based on the VSLIP-SL template model

$$
\begin{aligned}
h_1 &= y_{\mathrm{M}} - y^*_{\mathrm{M}}(x_{\mathrm{M}}) \\
h_2 &= x_{\mathrm{sw}} - x^*_{\mathrm{sw}}(x_{\mathrm{M}}) \\
h_3 &= y_{\mathrm{sw}} - y^*_{\mathrm{sw}}(x_{\mathrm{M}})
\end{aligned} \tag{3.13}
$$

where the terms marked with "$*$" are the reference trajectories.

To derive the controller, exact linearization via feedback for multi-input multi-output systems [18] will be used. Variable stiffness inputs for the single stance phase can be calculated as:

$$
\bar{\boldsymbol{u}}_{\mathrm{ss}} = \begin{bmatrix} \bar{u}_1 \\ \bar{u}_2 \\ \bar{u}_3 \end{bmatrix} = \boldsymbol{E}\boldsymbol{A}_{\mathrm{ss}}^{-1} \begin{bmatrix} -L^2_{f_{\mathrm{ss}}} h_1 - \kappa_d L_{f_{\mathrm{ss}}} h_1 - \kappa_p h_1 \\ -L^2_{f_{\mathrm{ss}}} h_2 - \kappa_d L_{f_{\mathrm{ss}}} h_2 - \kappa_p h_2 \\ -L^2_{f_{\mathrm{ss}}} h_3 - \kappa_d L_{f_{\mathrm{ss}}} h_3 - \kappa_p h_3 \end{bmatrix}, \tag{3.14}
$$

$$
\boldsymbol{A}_{\mathrm{ss}} = \begin{bmatrix} L_{g_1} L_{f_{\mathrm{ss}}} h_1 & L_{g_2} L_{f_{\mathrm{ss}}} h_1 & L_{g_3} L_{f_{\mathrm{ss}}} h_1 \\ L_{g_1} L_{f_{\mathrm{ss}}} h_2 & L_{g_2} L_{f_{\mathrm{ss}}} h_2 & L_{g_3} L_{f_{\mathrm{ss}}} h_2 \\ L_{g_1} L_{f_{\mathrm{ss}}} h_3 & L_{g_2} L_{f_{\mathrm{ss}}} h_3 & L_{g_3} L_{f_{\mathrm{ss}}} h_3 \end{bmatrix}, \tag{3.15}
$$

where $L_f h_i$, $L^2_f h_i$ and $L_{g_i} L_f h_i$ denote the Lie-derivatives of the output functions along the vector fields defined in (3.12) and κ_p, κ_d are constant controller gains. $\boldsymbol{A}_{\mathrm{ss}}$ matrix has singularities when the swing leg and swing foot springs are at their free lengths ($r = r_0$, $\theta = \theta_0$). This is why a regularization matrix $\boldsymbol{E}$ is needed. It is given by:

$$
\boldsymbol{E} = \begin{bmatrix} \frac{1}{\epsilon_1 + 1} & 0 & 0 \\ 0 & \frac{1}{\epsilon_2 + 1} & 0 \\ 0 & 0 & \frac{1}{\epsilon_3 + 1} \end{bmatrix}. \tag{3.16}
$$

The terms in the regularization matrix were chosen as $\epsilon_{1,2,3} = 10^{-7}/\det(\boldsymbol{A}_{\mathrm{ss}})$ in this study. We follow the same procedure for the double stance phase. First, the state is defined as $\boldsymbol{z}_{\mathrm{ds}} = [x_{\mathrm{CoM}}, y_{\mathrm{CoM}}, \dot{x}_{\mathrm{CoM}}, \dot{y}_{\mathrm{CoM}}]^T$. Then, the dynamic equation (3.2) is written in the state space form as:

$$
\dot{\boldsymbol{z}}_{\mathrm{ds}} = \boldsymbol{f}_{\mathrm{ds}}(\boldsymbol{z}_{\mathrm{ds}}) + \boldsymbol{g}_{\mathrm{ds}}(\boldsymbol{z}_{\mathrm{ds}}) \bar{\boldsymbol{u}}_{\mathrm{ds}}. \tag{3.17}
$$

where $\boldsymbol{g}_{\mathrm{ds}} = [\boldsymbol{g}_4, \boldsymbol{g}_5]^T \in \mathbb{R}^{4\times 2}$.
In the double stance phase there are two variable stiffness springs, which means we have two inputs. We want to control the vertical position and the horizontal velocity of CoM. To achieve this, outputs should be chosen as:

$$
\begin{aligned}
h_4 &= y_{\mathrm{CoM}} - y^*_{\mathrm{CoM}}(x_{\mathrm{CoM}}) \\
h_5 &= \dot{x}_{\mathrm{CoM}} - \dot{x}^*_{\mathrm{CoM}}(x_{\mathrm{CoM}})
\end{aligned} \tag{3.18}
$$

Again, by using the exact linearization via feedback, the controller for the double stance phase can be found as:

$$
\bar{\boldsymbol{u}}_{\mathrm{ds}} = \begin{bmatrix} \bar{u}_4 \\ \bar{u}_5 \end{bmatrix} = \boldsymbol{A}_{\mathrm{ds}}^{-1} \begin{bmatrix} -L^2_{f_{\mathrm{ds}}} h_4 - \kappa_d L_{f_{\mathrm{ds}}} h_4 - \kappa_p h_4 \\ -L_{f_{\mathrm{ds}}} h_5 - \kappa_v h_5 \end{bmatrix}, \tag{3.19}
$$

$$\boldsymbol{A}_{\mathrm{ds}} = \begin{bmatrix} L_{g_4} L_{f_{\mathrm{ds}}} h_4 & L_{g_5} L_{f_{\mathrm{ds}}} h_4 \\ L_{g_4} h_5 & L_{g_5} h_5 \end{bmatrix}. \tag{3.20}$$

Note that the second row of $\boldsymbol{A}_{\mathrm{ds}}$ is different since h_5 has a relative degree of 1 and h_1, h_2, h_3, h_4 have relative degrees of 2. $\boldsymbol{A}_{\mathrm{ds}}$ does not have singularities when both spring legs are in compression, which is already a necessary condition for a feasible SLIP-SL gait.

The reference trajectories, denoted as $\boldsymbol{z}^*_{\mathrm{ss}} = [\bar{\boldsymbol{q}}^*, \dot{\bar{\boldsymbol{q}}}^*]$ and $\boldsymbol{z}^*_{\mathrm{ds}} = [x^*_{\mathrm{CoM}}, y^*_{\mathrm{CoM}}, \dot{x}^*_{\mathrm{CoM}}, \dot{y}^*_{\mathrm{CoM}}]$, are provided with an asterisk symbol to indicate that they represent the natural trajectories of the SLIP-SL model, which do not involve any external inputs. These reference trajectories are formulated as functions of x_{CoM}, which increase monotonically, facilitating the computation of Lie derivatives.

In Visser et al.'s previous work [15], Fourier series expansion approximations were utilized to generate reference terms based on numerical solutions of the template model. However, this approach led to minor periodic deviations from the actual trajectory, even in the absence of disturbances. In this chapter, we adopted a different approach. We directly incorporated the numerical solution data obtained from the SLIP-SL model into the controller using MATLAB's symbolic toolbox, resulting in improved outcomes. For a detailed explanation of the natural trajectory of the SLIP-SL model and its generation process, we refer readers to [17].

3.2.2 5-Link Bipedal Robot Model Controller

The controller designed for the 5-link model faces the challenge of tracking the template model while incorporating its variable stiffness responses. However, it is important to note that the 5-link model does not possess any variable stiffness springs or compliant elements. Controller diagram can be seen in Figure 3.4.

Given this constraint, the controller must utilize the available actuation torques effectively to emulate the variable stiffness responses of the template model. This requires careful design and coordination of the control inputs to approximate the desired variable stiffness behavior. By appropriately modulating the torques applied to the joints of the 5-link model, we aim to replicate the variable stiffness responses observed in the template model, thereby achieving the desired walking patterns and behaviors.

For this purpose, we decided to use a joint trajectory tracking scheme via feedback linearization. Feedback linearization was chosen because it is easy to implement and consistent with the task. Keeping the equation of motion of the 5-link model in the single stance phase (3.3) in mind, if we choose the inputs as:

$$\boldsymbol{u} = \boldsymbol{S}^{-1}(\boldsymbol{M}\ddot{\boldsymbol{q}}^*_{\mathrm{ss}} + \boldsymbol{H}), \tag{3.21}$$

we can have $\ddot{\boldsymbol{q}} = \ddot{\boldsymbol{q}}^*_{\mathrm{ss}}$. Choosing

$$\ddot{\boldsymbol{q}}^*_{\text{ss}} = \begin{bmatrix} \ddot{\theta}^*_1 + \text{K}_{\text{p1}}(\theta^*_1 - \theta_1) + \text{K}_{\text{d1}}(\dot{\theta}^*_1 - \dot{\theta}_1) \\ \ddot{\theta}^*_2 + \text{K}_{\text{p2}}(\theta^*_2 - \theta_2) + \text{K}_{\text{d2}}(\dot{\theta}^*_2 - \dot{\theta}_2) \\ \ddot{\theta}^*_3 + \text{K}_{\text{p3}}(\theta^*_3 - \theta_3) + \text{K}_{\text{d3}}(\dot{\theta}^*_3 - \dot{\theta}_3) \\ \ddot{\theta}^*_4 + \text{K}_{\text{p4}}(\theta^*_4 - \theta_4) + \text{K}_{\text{d4}}(\dot{\theta}^*_4 - \dot{\theta}_4) \\ \ddot{\theta}^*_5 + \text{K}_{\text{p5}}(\theta^*_5 - \theta_5) + \text{K}_{\text{d5}}(\dot{\theta}^*_5 - \dot{\theta}_5) \end{bmatrix}, \tag{3.22}$$

will allow us to track the desired joint trajectories θ^*_i, $\dot{\theta}^*_i$ and the inclusion of $\ddot{\theta}^*_i$ terms are quite crucial since they will allow us to incorporate the variable stiffness responses that the VSLIP-SL controller gives against disturbances. $\ddot{\theta}^*_i$ are calculated using the VSLIP-SL dynamics (Equations (3.1) and (3.2)) and the decoder.

For the double stance phase, the controller needs to be modified. First, we will reduce the order of the dynamical model since both legs are on the ground and are stationary. The velocity of the hip can be calculated equivalently using the stance leg and swing leg gen. coordinates as:

$$\dot{\boldsymbol{q}}_{\text{hip}} = \boldsymbol{J}_{\text{hip,sw}}(\boldsymbol{q}_{\text{sw}})\dot{\boldsymbol{q}}_{\text{sw}} = \boldsymbol{J}_{\text{hip,st}}(\boldsymbol{q}_{\text{st}})\dot{\boldsymbol{q}}_{\text{st}}, \tag{3.23}$$

where $\boldsymbol{J}_{\text{hip},\,i}(\boldsymbol{q}_i)$ is the Jacobian matrix that maps $\dot{\boldsymbol{q}}_{\text{st}} = [\dot{\theta}_1, \dot{\theta}_2]^T$ and $\dot{\boldsymbol{q}}_{\text{sw}} = [\dot{\theta}_3, \dot{\theta}_4]^T$ to the hip velocity. Equation (3.23) can be used to obtain:

$$\dot{\boldsymbol{q}} = \underbrace{\begin{bmatrix} \boldsymbol{J}^{-1}_{\text{hip,st}}\boldsymbol{J}_{\text{hip,sw}} & 0_{2\times1} \\ I_{2\times2} & 0_{2\times1} \\ 0_{1\times2} & 1 \end{bmatrix}}_{\boldsymbol{\Gamma}(\boldsymbol{q})\in\mathbb{R}^{5\times3}} \underbrace{\begin{bmatrix} \dot{\boldsymbol{q}}_{\text{sw}} \\ \dot{\theta}_5 \end{bmatrix}}_{\dot{\boldsymbol{q}}_a}. \tag{3.24}$$

By taking the time derivative of (3.24), then substituting the resulting $\ddot{\boldsymbol{q}}$ and $\dot{\boldsymbol{q}}$ terms into (3.3) and multiplying with $\boldsymbol{\Gamma}^T$ from the left, the following expression can be obtained:

$$\boldsymbol{M}_a(\boldsymbol{q})\ddot{\boldsymbol{q}}_a + \boldsymbol{H}_a(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{u}_a, \tag{3.25}$$

where $\boldsymbol{M}_a = \boldsymbol{\Gamma}^T\boldsymbol{M}\boldsymbol{\Gamma}$, $\boldsymbol{H}_a = \boldsymbol{\Gamma}^T(\boldsymbol{M}\dot{\boldsymbol{\Gamma}}\dot{\boldsymbol{q}}_a + \boldsymbol{H})$ and $\boldsymbol{u}_a = \boldsymbol{\Gamma}^T\boldsymbol{S}\boldsymbol{u}$, is the reduced order model for the double stance phase. If inputs are chosen as:

$$\boldsymbol{u}_a = \boldsymbol{M}_a\ddot{q}^*_{\text{ds}}\boldsymbol{H}_a, \tag{3.26}$$

we can have $\boldsymbol{q}_a = \ddot{\boldsymbol{q}}_{\text{ds}}$. Choosing

$$\ddot{\boldsymbol{q}}^*_{\text{ds}} = \begin{bmatrix} \ddot{\theta}^*_3 + K_p(\theta^*_3 - \theta_3) + K_d(\dot{\theta}^*_3 - \dot{\theta}_3) \\ \ddot{\theta}^*_4 + K_p(\theta^*_4 - \theta_4) + K_d(\dot{\theta}^*_4 - \dot{\theta}_4) \\ \ddot{\theta}^*_5 + K_p(\theta^*_5 - \theta_5) + K_d(\dot{\theta}^*_5 - \dot{\theta}_5) \end{bmatrix}, \tag{3.27}$$

will allow us to track the desired trajectories for $\theta_{3,4,5}$. To find the proper inputs corresponding to the $\theta_{1,2}$, we can use the relation $\boldsymbol{u}_a = \boldsymbol{\Gamma}^T\boldsymbol{S}\boldsymbol{u}$:

$$\boldsymbol{u} = \boldsymbol{S}^{-1}\left((\boldsymbol{\Gamma}^T)^{+\boldsymbol{W}}\boldsymbol{u}_a\right), \tag{3.28}$$

where $(\boldsymbol{\Gamma}^T)^{+\boldsymbol{W}} = \boldsymbol{W}^{-1}\boldsymbol{\Gamma}(\boldsymbol{\Gamma}^T\boldsymbol{W}^{-1}\boldsymbol{\Gamma})^{-1}$ is the weighted matrix pseudoinverse operation. $\boldsymbol{W} \in \mathbb{R}^{5\times5}$ matrix can be used to penalize high input torques such as the ankle torque but we selected it as identity matrix for this chapter.

3.2.3 Encoder-Decoder

Now that we have defined the controllers for VSLIP-SL and the 5-link model, the remaining problem is to connect them together. We will achieve this through the reference trajectories indicated by "*" in equations (3.22) and (3.27).

In order to obtain $\ddot{\theta}_i^*$, $\dot{\theta}_i^*$, θ_i^* terms from the reference VSLIP-SL trajectories and variable stiffness responses $\bar{\boldsymbol{u}}$, we first need to find the equivalent VSLIP-SL states for the 5-link model at that instant. We will call this process “encoding".

For the single stance phase, this will be achieved by solving the

$$\mathbf{F}_{\boldsymbol{x}}^{\text{enc}} = \begin{bmatrix} x_{\text{CoM}}(\boldsymbol{q}) - \bar{x}_{\text{CoM}}(\bar{\boldsymbol{q}}) \\ y_{\text{CoM}}(\boldsymbol{q}) - \bar{y}_{\text{CoM}}(\bar{\boldsymbol{q}}) \\ x_{\text{sw}}(\boldsymbol{q}) - \bar{x}_{\text{sw}}(\bar{\boldsymbol{q}}) \\ y_{\text{sw}}(\boldsymbol{q}) - \bar{y}_{\text{sw}}(\bar{\boldsymbol{q}}) \end{bmatrix} = 0, \tag{3.29}$$

set of equations to find appropriate $\bar{\boldsymbol{q}} = [x_{\text{M}}, y_{\text{M}}, \theta, r]^T$ values.

Then using the resulting $\bar{\boldsymbol{q}}$, we solve

$$\mathbf{F}_{\dot{\boldsymbol{x}}}^{\text{enc}} = \begin{bmatrix} \dot{x}_{\text{CoM}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\bar{x}}_{\text{CoM}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \\ \dot{y}_{\text{CoM}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\bar{y}}_{\text{CoM}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \\ \dot{x}_{\text{sw}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\bar{x}}_{\text{sw}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \\ \dot{y}_{\text{sw}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\bar{y}}_{\text{sw}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \end{bmatrix} = 0, \tag{3.30}$$

for $\dot{\bar{\boldsymbol{q}}}$. Solving $\mathbf{F}_{\boldsymbol{x}}^{\text{enc}}$ and $\mathbf{F}_{\dot{\boldsymbol{x}}}^{\text{enc}}$ gives us the equivalent VSLIP-SL model to the current state of the 5-link model. Using $\bar{\boldsymbol{q}}$ and $\dot{\bar{\boldsymbol{q}}}$, we can get the variable stiffness response $\bar{\boldsymbol{u}}_{\text{ss}}$ of the VSLIP-SL controller (3.14).

For the double stance phase, we can directly use $x_{\text{CoM}}(\boldsymbol{q})$, $y_{\text{CoM}}(\boldsymbol{q})$, $\dot{x}_{\text{CoM}}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, $\dot{y}_{\text{CoM}}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ values of the 5-link model because these are already the states of VSLIP-SL in double stance phase. These can be used in (3.19) to obtain the $\bar{\boldsymbol{u}}_{\text{ds}}$ values.

Next step is obtaining the desired joint trajectories using these values. We call this process “decoding". By solving the sets of nonlinear equations

$$\mathbf{F}_{\boldsymbol{x}}^{\text{dec}} = \begin{bmatrix} \bar{x}_{\text{CoM}}^* - x_{\text{CoM}}(\boldsymbol{q}^*) \\ \bar{y}_{\text{CoM}}^* - y_{\text{CoM}}(\boldsymbol{q}^*) \\ \bar{x}_{\text{sw}}^* - x_{\text{sw}}(\boldsymbol{q}^*) \\ \bar{y}_{\text{sw}}^* - y_{\text{sw}}(\boldsymbol{q}^*) \\ \theta_5^* - \pi/2 \end{bmatrix} = 0, \tag{3.31}$$

$$\mathbf{F}_{\dot{\boldsymbol{x}}}^{\text{dec}} = \begin{bmatrix} \dot{\bar{x}}^*_{\text{CoM}} - \dot{x}_{\text{CoM}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*) \\ \dot{\bar{y}}^*_{\text{CoM}} - \dot{y}_{\text{CoM}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*) \\ \dot{\bar{x}}^*_{\text{sw}} - \dot{x}_{\text{sw}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*) \\ \dot{\bar{y}}^*_{\text{sw}} - \dot{y}_{\text{sw}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*) \\ (\dot{\theta}_5^*)^2 \end{bmatrix} = 0, \tag{3.32}$$

$$\mathbf{F}_{\ddot{\boldsymbol{x}}}^{\text{dec}} = \begin{bmatrix} \ddot{\bar{x}}_{\text{CoM}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}, \bar{\boldsymbol{u}}_{\text{ss,ds}}) - \ddot{x}_{\text{CoM}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*, \ddot{\boldsymbol{q}}^*) \\ \ddot{\bar{y}}_{\text{CoM}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}, \bar{\boldsymbol{u}}_{\text{ss,ds}}) - \ddot{y}_{\text{CoM}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*, \ddot{\boldsymbol{q}}^*) \\ \ddot{\bar{x}}_{\text{sw}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}, \bar{\boldsymbol{u}}_{\text{ss,ds}}) - \ddot{x}_{\text{sw}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*, \ddot{\boldsymbol{q}}^*) \\ \ddot{\bar{y}}_{\text{sw}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}, \bar{\boldsymbol{u}}_{\text{ss,ds}}) - \ddot{y}_{\text{sw}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*, \ddot{\boldsymbol{q}}^*) \\ (\ddot{\boldsymbol{q}}^*)^T \ddot{\boldsymbol{q}}^* \end{bmatrix} = 0, \tag{3.33}$$

consecutively, we can obtain the

$$\boldsymbol{q}^* = [\theta_1^*, \theta_2^*, \theta_3^*, \theta_4^*, \theta_5^*]^T, \tag{3.34}$$

$\dot{\boldsymbol{q}}^*$ and $\ddot{\boldsymbol{q}}^*$ to be used in the joint angle tracking controller for the 5-link model (3.21). In the double stance phase, $\bar{x}^*_{\text{sw}}$ and $\bar{y}^*_{\text{sw}}$ are set to the current position of the robot's foot on the ground and $\dot{\bar{x}}^*_{\text{sw}}$, $\dot{\bar{y}}^*_{\text{sw}}$, $\ddot{\bar{x}}_{\text{sw}}$, $\ddot{\bar{y}}_{\text{sw}}$ are set to 0. The last row of (3.33) is to minimize the control effort.

We use the `fsolve` function of Matlab with the "Levenberg-Marquardt" option for solving sytems of nonlinear equations in this chapter.

3.3 Results and Discussion

In this sections, we share the simulation results for the proposed VSLIP-SL controller and the 5-link model controller that is based on it. We conducted simulation studies in Matlab SIMULINK environment with variable step `ode45` solver with a max step size of 1e-3 and an absolute tolerance of 1e-8.

3.3.1 VSLIP-SL Results

The parameters for the VSLIP-SL model and 5-link model are given in Table 3.1 and controller parameters are given in Table 3.3.

Table 3.1: VSLIP-SL Parameters

5-Link Model		
$l_1 = l_4 : 0.55[m]$	$l_2 = l_3 : 0.50[m]$	$l_5 : 0.30[m]$
$m_1 = m_4 : 4.75[kg]$	$m_2 = m_3 : 5.25[kg]$	$m_5 : 60[kg]$
$I_i = m_i l_i^2/12[kg \cdot m^2], i = 1, 2, 3, 4, 5$		

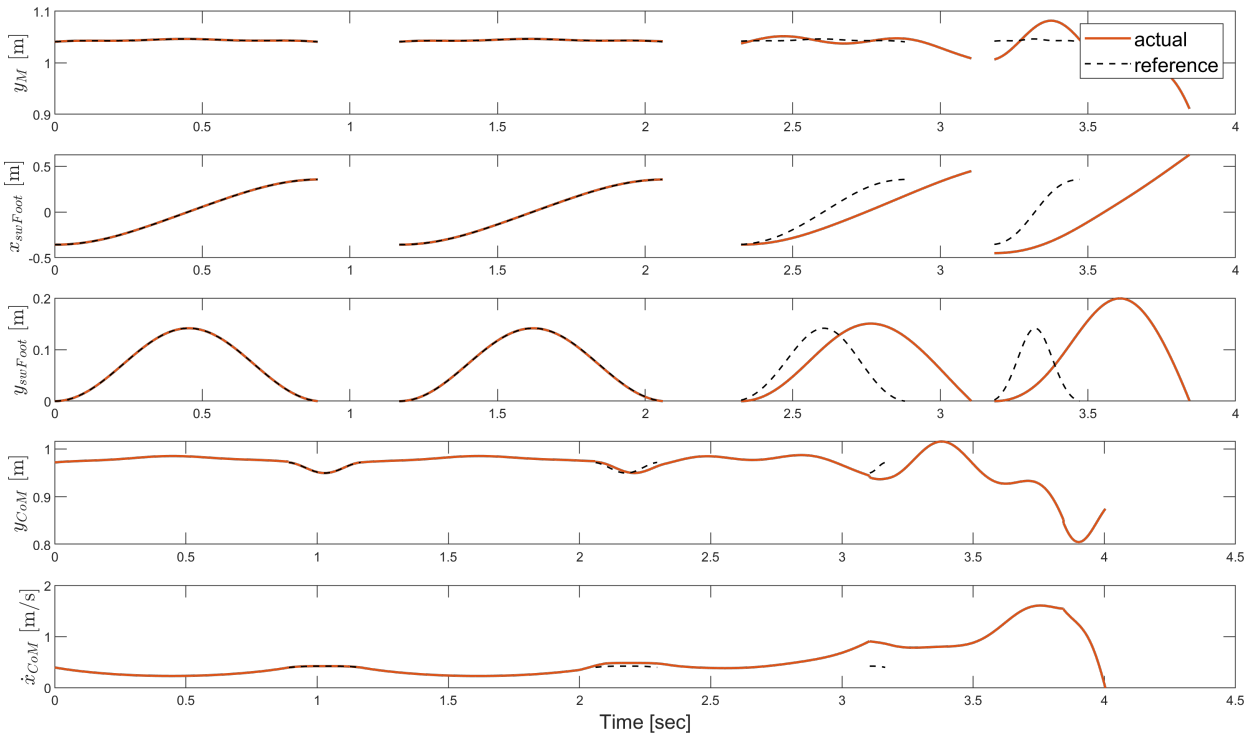


Figure 3.5: SLIP-SL trajectory tracking results when a disturbance force of $F_{dist} = [50, 50]$ [N] is applied between 2.0 secs and 2.1 secs

Table 3.2: 5-link Bipedal Robot Model Parameters

5-Link Model		
$l_1 = l_4 : 0.55[m]$	$l_2 = l_3 : 0.50[m]$	$l_5 : 0.30[m]$
$m_1 = m_4 : 4.75[kg]$	$m_2 = m_3 : 5.25[kg]$	$m_5 : 60[kg]$
$I_i = m_i l_i^2/12[kg \cdot m^2], i = 1, 2, 3, 4, 5$		

SLIP-SL was the original model with constant spring stifnesses. Figure 3.5 shows the trajectory tracking results for the VSLIP-SL model when a disturbance force of $F_{dist} = [50, 50]$ [N] is applied between 2.0 secs and 2.1 secs. We can see that it can't handle the external disturbance and falls.

Figure 3.7 shows the trajectory tracking results and Figure 3.6 shows how the stiffness values have changed for the VSLIP-SL model when a disturbance force of $F_{dist} = [50, 50]$ [N] is applied between 2.0 secs and 2.1 secs. We can see that it can recover after the external push and converge back to its original trajectory.

3.3.2 5-link Bipedal Robot Results

The parameters for the 5-link model are given in Table 3.2 and controller parameters are given in Table 3.3.

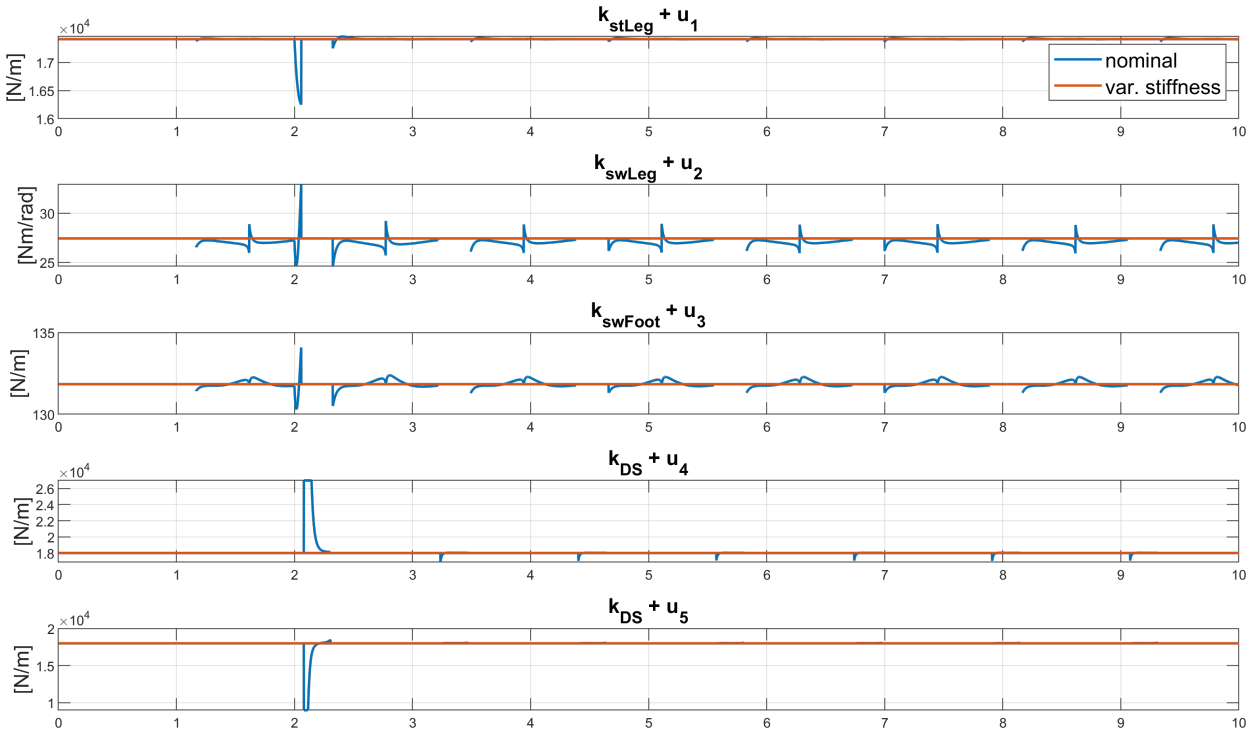


Figure 3.6: VSLIP-SL spring stiffness changes when a disturbance force of $\mathrm{F}_{\mathrm{dist}} = [50, 50]$ [N] is applied between 2.0 secs and 2.1 secs

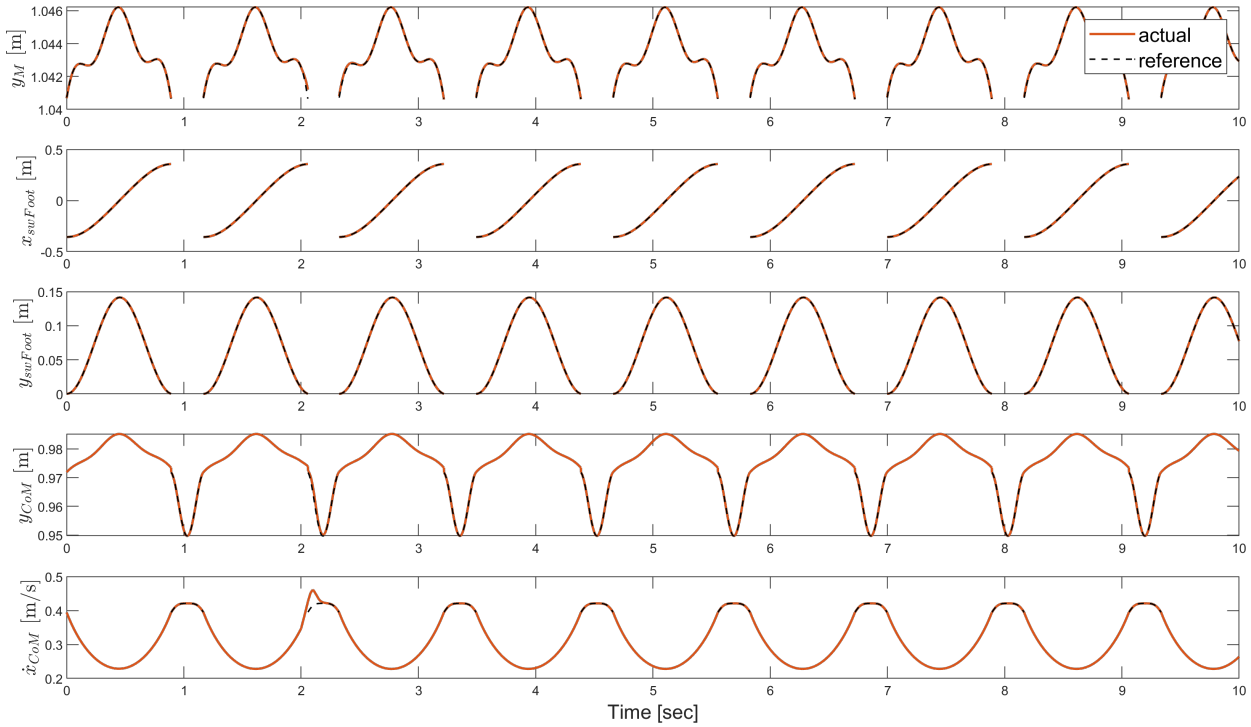


Figure 3.7: VSLIP-SL trajectory tracking results when a disturbance force of $\mathrm{F}_{\mathrm{dist}} = [50, 50]$ [N] is applied between 2.0 secs and 2.1 secs

To test the robustness of the controller, a disturbance force of $[-100, 0]^T$ N is applied between 10.0 secs and 10.1 secs at the hip. The controller was able to handle the disturbance and the 5-link model was able to continue its gait without falling.

Figure 3.8 shows the trajectory tracking results. We can see that the proposed controllers performance in tracking the CoM and swing foot trajectory is quite good, even under the effect of the disturbance. Inputs during this gait are given in Figure 3.10 and the snapshots from it can be seen in Figure 3.9.

Cost of transport [5] for the reference SLIP-SL gait was 0.1907 with an average velocity of 0.3061 [m/s]. The cost of transport for the 5-link model's nominal trajectory using the proposed controller is 0.3364 with an average velocity of 0.3116 [m/s]. A slightly higher cost of transport is reasonable since SLIP-SL model does not have a body that it needs to keep upright. We constrained the inputs to be between $[-200, 200]$ [Nm]. The zero moment point (ZMP) [2] stays within the virtual foot (ZMP remains in the $[-0.10, 0.19]$ [m] region). Average elapsed time for encoding is 3.3e-5 secs with a maximum of 1.9e-4 secs, and the average elapsed time for decoding is 4.6e-4 secs with a maximum of 2.1e-3 secs.

Figure 3.11 shows the disturbance test results. We tested our controller with different magnitudes of external forces shown on the figure (F_x and F_y). The area represent that the bipedal robot was able to overcome that specific disturbance force. Left figure shows the results when the VSLIP-SL controller is disabled i.e., set

$$\ddot{\theta}_i^* = 0, \tag{3.35}$$

in equations (3.22) and (3.27). Figure on the right shows the result of our proposed controller. We can see that the area on the figure has significantly increased. This shows that our proposed controller increases the robustness compared to the nominal controller.

Table 3.3: Controller Parameters

$K_{p1} = 835$	$K_{d1} = 100$	$K_{p2} = 370$	$K_{d2} = 60$
$K_{p3} = 445$	$K_{d3} = 110$	$K_{p4} = 550$	$K_{d4} = 310$
$K_{p5} = 375$	$K_{d5} = 140$	$\kappa_p = 200$	$\kappa_d = 40$ $\kappa_v = 40$

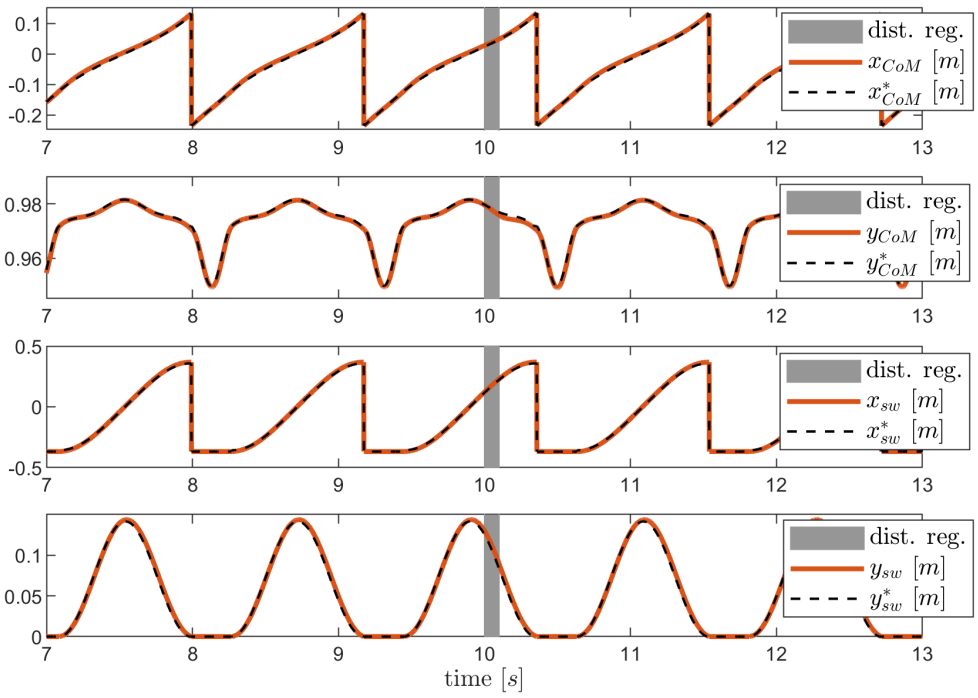


Figure 3.8: Trajectory tracking results of the 5-link model. A disturbance force is applied during the gray region

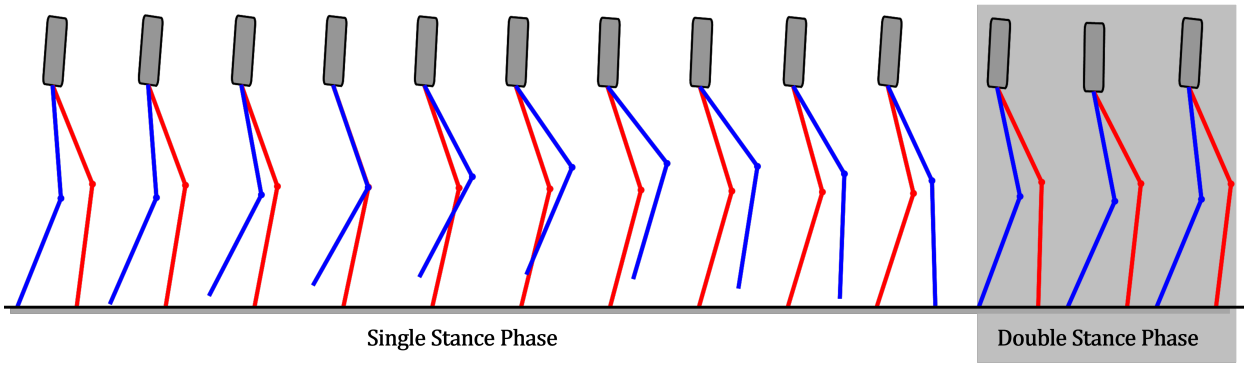


Figure 3.9: Snapshots from the 5-link model's gait.

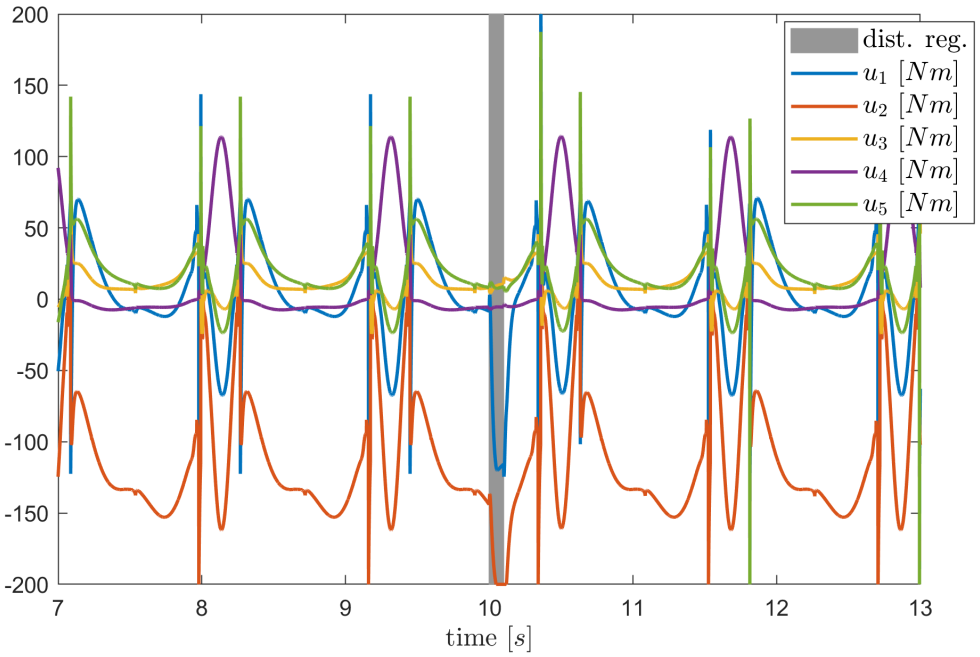
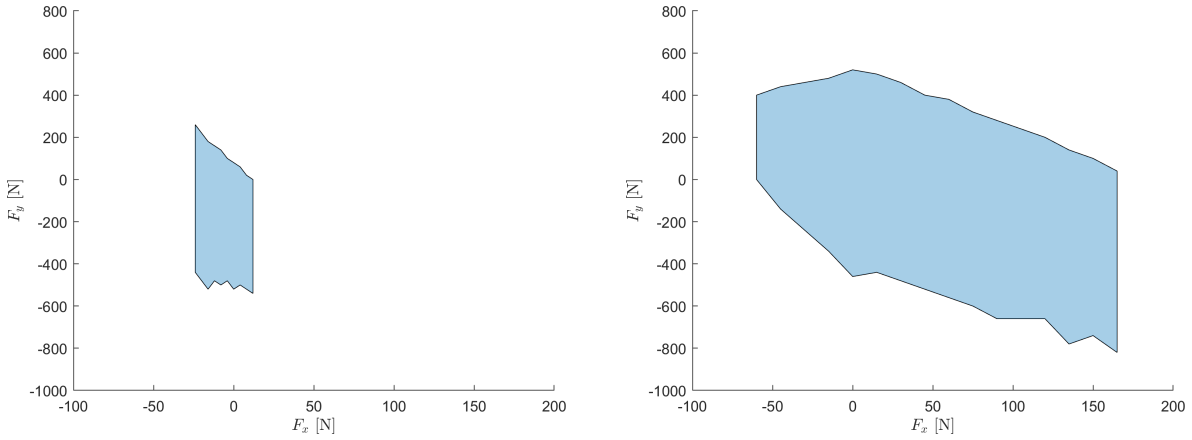


Figure 3.10: Input torques of the 5-link model for tracking the VSLIP-SL trajectory.



(a) Without VSLIP-SL controller

(b) With VSLIP-SL controller

Figure 3.11: Disturbance test results where a area represent that the biped robot was able to overcome that external disturbance force

3.4 Conclusion

In this chapter, we first added variable stiffness to SLIP-SL template model to make it robust against disturbances. We called this new model VSLIP-SL. The significance of the proposed model is that it can provide the reference CoM trajectory as well as the reference swing foot trajectory while rejecting disturbances via its variable stiffness springs.

Subsequently, we proposed a controller for a 5-link fully actuated bipedal robot model, which enabled the utilization of the variable stiffness responses from the VSLIP-SL model. The main challenge lay in translating these variable stiffness responses into a model lacking any compliant elements. To address this challenge, we introduced an encoder-decoder structure within the controller. This structure facilitated the mapping of the variable stiffness responses to the actuation inputs of the 5-link robot model.

Through the implementation of the proposed controller, we successfully achieved the tracking of the reference SLIP-SL trajectories while effectively dealing with external disturbance forces. This demonstrates the capability of the controller to leverage the variable stiffness responses of the VSLIP-SL model, enabling the bipedal robot to exhibit robust and adaptive walking behavior.

Chapter 4

Effects of Passive Biarticular Muscles on Walking Performance for Bipedal Robots

Enhancing the efficiency of bipedal robots has been a continued focus in research, and drawing inspiration from human anatomy has proven to be a valuable approach. While the human locomotion system utilizes both monoarticular (acting on a single joint) and biarticular (acting on multiple joints) muscles in a coordinated manner during walking, the incorporation of biarticular muscles remains relatively unexplored in current legged robotics applications.

Biarticular muscles play a significant role in human walking by providing mechanical advantages, energy optimization, and improved force transmission across multiple joints. By exploiting the capabilities of biarticular muscles, bipedal robots have the potential to achieve greater efficiency and performance, akin to their human counterparts.

Monoarticular muscles are connected to two links and can drive a single joint. On the other hand, biarticular muscles are connected to two links separated by a third one and they can drive two joints at the same time [19]. Biarticular muscles in the human leg are shown in a simplified manner in Figure 4.1. Rectus Femoris (RF) and hamstrings (HA) are connected between the upper body and the shank while gastrocnemius is connected between the thigh and the foot.

Studies have highlighted the benefits of incorporating biarticular muscles into both robotic and template walking systems. These studies have demonstrated the advantages of utilizing biarticular muscles in various scenarios.

In the study conducted by Oh et al. [20], a two-link manipulator was used to compare the performance of configurations with only monoarticular muscles versus configurations with mono and biarticular muscles when performing the same task. The results showed that the configuration with biarticular muscles required lower torques compared to the configuration with only monoarticular muscles. This highlights the mechanical advantages provided by

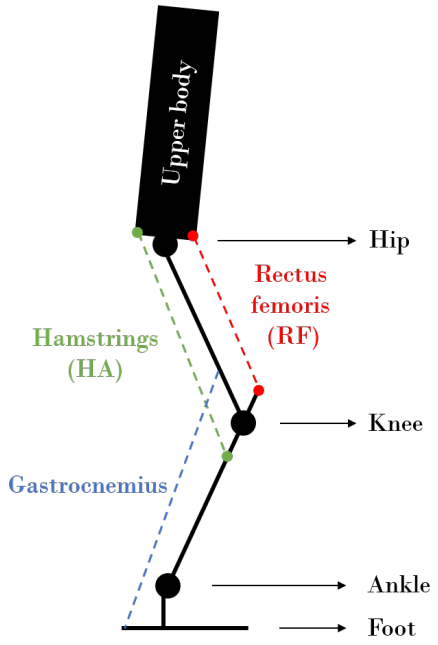


Figure 4.1: Simple human leg diagram showing the biarticular muscles

biarticular muscles, which can lead to improved energy efficiency and reduced torque requirements in robotic systems.

Another study by Hosoda et al. [21] investigated the use of biarticular muscles in an anthropomorphic jumping monopod with pneumatic actuators. They demonstrated that biarticular muscles contribute to joint coordination during the jumping motion. They were able to decouple trunk balancing and swinging tasks from the load carrying, allowing for more efficient and controlled jumping performance. This decoupling was only made possible because of the usage of series elastic actuated biarticular muscles.

Furthermore, the utilization of passive biarticular muscles for swing leg motion in a conceptual walking model was demonstrated in the work by Sharbafi et al. [8]. This study showcased the potential of biarticular muscles to realize the swinging motion of the leg, further emphasizing their importance in achieving natural and coordinated walking gaits.

We've discussed the importance of the mechanical design of the biped robots. Another very important component is the control method. It plays a crucial role in achieving stable and efficient walking gaits. Researchers have explored various approaches to address the control problem in bipedal locomotion.

In the study by Garofalo et al. [11], a simple template model called the bipedal spring-loaded inverted pendulum (B-SLIP) was employed as a reference for generating a walking gait. This template model provided a simplified representation of the walking dynamics, and a feedback linearization controller was utilized to achieve stable walking. The feedback linearization technique allowed for the translation of desired dynamics into control inputs,

facilitating the tracking of the B-SLIP reference trajectories.

Another approach was presented in the work by Yano et al. [3], where an optimal trajectory for a bipedal robot equipped with series elastic actuators was obtained using direct collocation methods [9]. By formulating the trajectory optimization problem and employing direct collocation, an optimal trajectory that satisfies certain performance criteria was derived. Subsequently, an input-output linearization control scheme was employed to track this optimal trajectory, ensuring accurate trajectory following and stability.

Another important aspect for walking bipedal robots is the control method. There are a variety of different approaches to this control problem. In [11], a simple template model called bipedal spring loaded inverted pendulum was used as a reference to achieve a walking gait with the help of a feedback linearization controller. In [3], optimal trajectory of a bipedal robot with series elastic actuators was found using direct collocation methods [9], then this trajectory was tracked using input-output linearization control.

In this chapter, our focus was to examine the impact of passive biarticular muscles on the performance of bipedal walking. To achieve this, we employed direct collocation optimization methods to compare the performance of two models: the nominal model, which only consists of monoarticular actuators, and the BA model, which incorporates passive biarticular muscles (RF and HA).

We assessed the effects of passive biarticular muscles using three different criteria: walking efficiency, walking speed, and minimum actuator torque requirements. By conducting analyses, we aimed to determine the advantages and potential benefits of incorporating passive biarticular muscles in bipedal locomotion.Furthermore, we employ optimization methods to find suitable parameters for the biarticular springs for the criteria in question. We also aim to provide a useful tool for determining optimal parameters for biarticular springs, further improving the overall performance of the bipedal walking system.

Subsequently, we propose a controller that enables the achievement of a stable walking gait with the bipedal robot model incorporating biarticular springs. The proposed controller adopts a feedback-plus-feedforward structure, which facilitates the tracking of both the reference trajectories and the optimal inputs obtained through the optimization process. We then test the performance of the proposed controller through simulation experiments.

This chapter is organized as follows: Section 4.1 deals with the modeling of the nominal and BA models, in Section 4.2 simultaneous trajectory and parameter optimization methods are explained and Section 4.3 introduces the proposed controller. Finally numerical simulation results are presented and discussed in Section 4.4, followed by the conclusion in Section 4.5.

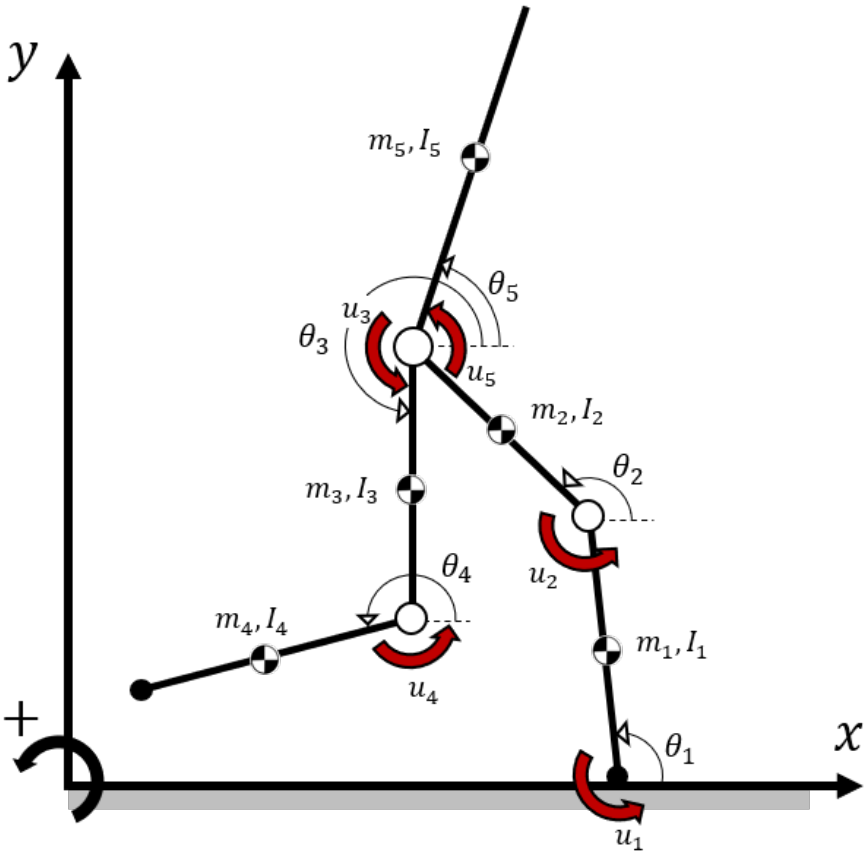


Figure 4.2: Nominal Bipedal Robot Model

4.1 Systems and Modeling

In this section, dynamics of the nominal bipedal model and the model with passive biarticular muscles (BA model) will be introduced.

4.1.1 Nominal model

The nominal robot model can be seen in Figure 4.2. This planar model consists of 5 links which are connected to each other by revolute joints. It has five actuators, two at knees, two at the hip and one on the “ankle". Dynamics of the nominal model can be written as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{S}\boldsymbol{u}, \tag{4.1}$$

where $\boldsymbol{q} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^T \in \mathbb{R}^5$ are the generalized coordinates, $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{5\times5}$ is the inertia matrix, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{5\times5}$ is the Coriolis and centrifugal terms matrix, $\boldsymbol{G}(\boldsymbol{q}) \in \mathbb{R}^5$ is the gravitational terms, $\boldsymbol{u} \in \mathbb{R}^5$ are the input torques and $\boldsymbol{S} \in \mathbb{R}^{5\times5}$ is the proper mapping matrix for the inputs. States of this system will be denoted as:

$$\boldsymbol{x} = [\boldsymbol{q}^{\mathrm{T}}, \dot{\boldsymbol{q}}^{\mathrm{T}}]^{\mathrm{T}}. \tag{4.2}$$

When the swing foot is above the ground, dynamics of the nominal model are described

by Equation (4.1). When the swing foot eventually comes into contact with the ground, a collision occurs. This collision results in a discontinuous change in the generalized coordinates and velocities but the overall position of the model is not affected. The collision can be modeled by assuming that an impulse force acts on the system at the moment of the impact. The discontinuous change in the generalized momentum due to this impulse force can be expressed as:

$$\boldsymbol{M}(\boldsymbol{q})\Delta\dot{\boldsymbol{q}} = \boldsymbol{J}_c^T \boldsymbol{\lambda}_{\text{impact}}, \tag{4.3}$$

where $\boldsymbol{J}_c \in \mathbb{R}^{2\times 5}$ is a constraint Jacobian matrix that maps the generalized velocities to the swing foot velocity, $\boldsymbol{\lambda}_{\text{impact}} \in \mathbb{R}^2$ is the impulse force and $\Delta\dot{\boldsymbol{q}}$ is the change in the generalized velocities. Assuming that the impact is inelastic, velocity of the swing foot touching the ground will become zero after the impact, which can be expressed as:

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}^+ = 0 \Leftrightarrow \boldsymbol{J}_c(\boldsymbol{q})\Delta\dot{\boldsymbol{q}} = -\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}^-, \tag{4.4}$$

where superscripts "$^-$" expresses the moment just before the impact and "$^+$" just after the impact. By solving equations (4.3) and (4.4) together we can get:

$$\dot{\boldsymbol{q}}^+ = (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{J}_c^T(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}\boldsymbol{J}_c)\dot{\boldsymbol{q}}^-, \tag{4.5}$$

which gives the joint velocities after the impact. Definitions of the legs also change i.e. the swing leg becomes the stance leg and vice versa.

Humans gait is two phased: a single stance phase and double stance phase. In the single stance phase, only one foot is on the ground and the other foot is "swinging", whereas in the double stance phase, both feet are in contact with the ground. In this chapter, we assume an instantaneous double stance phase i.e. swing leg and stance leg switch instantaneously at the moment of the impact and after the impact, swing leg lifts up from the ground without interaction. The reset map that maps generalized coordinates just before the impact to those just after the impact is given by:

$$\boldsymbol{x}^+ = f_H(\boldsymbol{x}^-). \tag{4.6}$$

4.1.2 Model with Biarticular Muscles (BA Model)

The model with passive biarticular muscles (springs) can be seen in Figure 4.3. This model will also be referred to as the "BA model" for brevity. BA model is the same as the nominal model but with the addition of lever arms attached to the torso and shank connected with 2 springs for each leg, representing rectus femoris (RF) and hamstrings (HA) muscles in humans. r_h and r_k respectively represent the lever arm lengths of the hip and knee. We will derive the torques resulting from RF and HA in the following parts.

Here, RF and HA are two unidirectional linear springs that can generate force only in one direction and they have the same set points. New variables representing the hip and knee angles ($\varphi_h^{\text{st}}(\boldsymbol{q})$, $\varphi_k^{\text{st}}(\boldsymbol{q})$, $\varphi_h^{\text{sw}}(\boldsymbol{q})$, $\varphi_k^{\text{sw}}(\boldsymbol{q})$) are indicated in Figure 4.3. Then, the potential

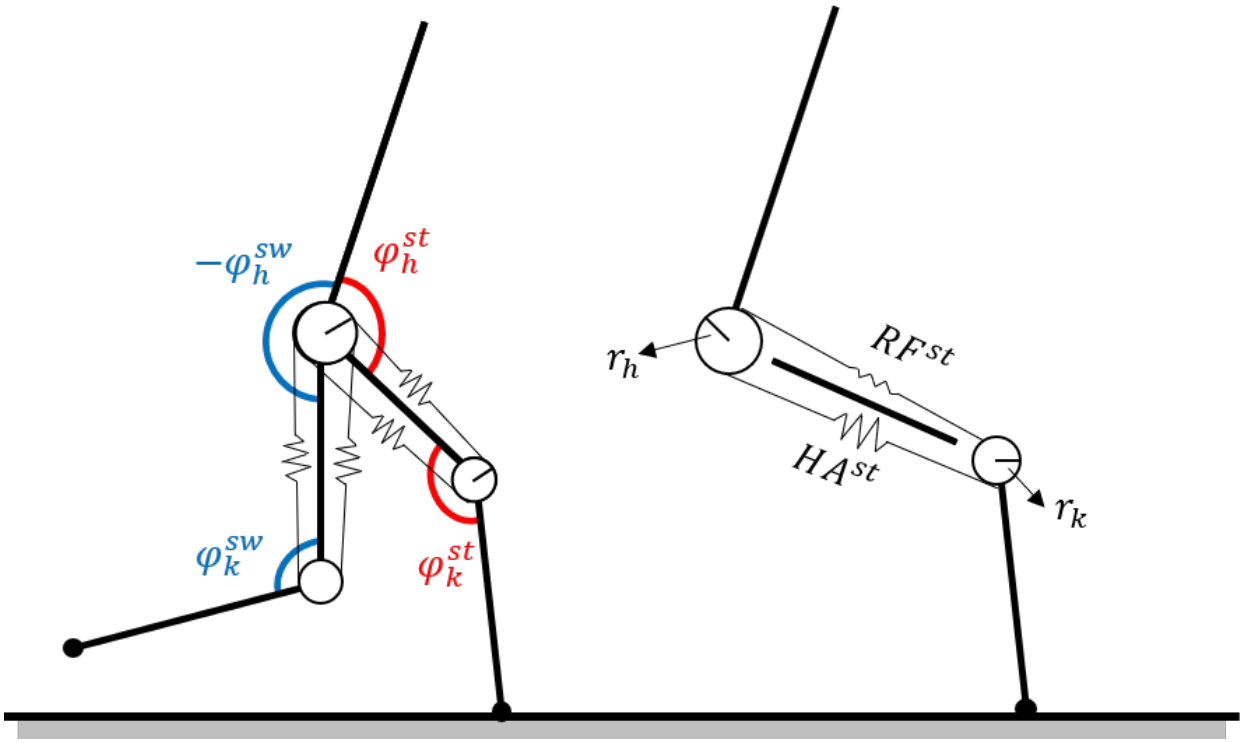


Figure 4.3: Model with biarticular muscles (BA Model)

energy stored in biarticular springs can be calculated in a similar manner to [22] as:

$$U_n = \frac{1}{2}\kappa \Delta l_n^2, \; n \in \{\text{sw}, \text{st}\}, \tag{4.7}$$

where κ $[N/m]$ is the stiffness of biarticular springs and Δl_n represents the deflection in the respective spring which can be calculated as:

$$\Delta l_n = r_h(\varphi_h^n - \varphi_{h0}) - r_k(\varphi_k^n - \varphi_{k0}), \; n \in \{\text{sw}, \text{st}\}. \tag{4.8}$$

Resulting generalized spring forces can be derived from the potential energy as:

$$\boldsymbol{\tau}_n = -\frac{\partial U_n}{\partial \boldsymbol{q}}, \;\; n \in \{\text{sw}, \text{st}\},$$

where

$$\boldsymbol{\tau}_{\text{st}} = \begin{bmatrix} 0 \\ -\kappa\, r_{\text{k}} \Delta l_{\text{st}} \\ 0 \\ 0 \\ -\kappa\, r_{\text{h}} \Delta l_{\text{st}} \end{bmatrix}, \; \boldsymbol{\tau}_{\text{sw}} = \begin{bmatrix} 0 \\ 0 \\ \kappa\, r_{\text{h}} \Delta l_{\text{sw}} \\ \kappa\, r_{\text{k}} \Delta l_{\text{sw}} \\ -\kappa\, r_{\text{h}} \Delta l_{\text{sw}} \end{bmatrix}, \tag{4.9}$$

We define the lever arm ratio as $r = r_h/r_k$ which is dimensionless. r was introduced because in the optimization section (Section 4.2), mechanical parameters of the biarticular muscles will be optimized simultaneously with the robot trajectory. By introducing a dimensionless

r, we can obtain a ratio rather than a dimension, which will be practical if an experimental system were to be designed and the actual dimensions of the lever arms can be adjusted with respect to the general mechanical design of the robot. We define new variables with this new notation as:

$$\bar{\kappa} = \kappa r_k^2,\ \overline{\Delta l}_i = \frac{\Delta l_i}{r_k}. \tag{4.10}$$

After proper manipulations, Equation (4.1.2) can be rewritten with the new variables as

$$\boldsymbol{\tau}_{st} = \begin{bmatrix} 0 \\ -\bar{\kappa}\overline{\Delta l}_{\mathrm{st}} \\ 0 \\ 0 \\ -\bar{\kappa}\overline{\Delta l}_{\mathrm{st}} r \end{bmatrix},\ \ \boldsymbol{\tau}_{sw} = \begin{bmatrix} 0 \\ 0 \\ \bar{\kappa}\overline{\Delta l}_{\mathrm{sw}} r \\ \bar{\kappa}\overline{\Delta l}_{\mathrm{sw}} \\ -\bar{\kappa}\overline{\Delta l}_{\mathrm{sw}} r \end{bmatrix}. \tag{4.11}$$

Finally, dynamics of the BA model can be written as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{S}\boldsymbol{u} + \boldsymbol{\tau}(\boldsymbol{q}), \tag{4.12}$$

where $\boldsymbol{q}$, $\boldsymbol{M}(\boldsymbol{q})$, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, $\boldsymbol{G}(\boldsymbol{q})$, $\boldsymbol{u}$, $\boldsymbol{S}$ are the same as in Equation (4.1) and

$$\boldsymbol{\tau}(\boldsymbol{q}) = \boldsymbol{\tau}_{\mathrm{sw}}(\boldsymbol{q}) + \boldsymbol{\tau}_{\mathrm{st}}(\boldsymbol{q}), \tag{4.13}$$

represents the torques generated by biarticular springs RF and HA. The biarticular spring configuration can be seen in Figure 4.4. Springs are installed in a parallel manner, there is no additional freedom between the BA spool and the joint. In other words, relative degree of this model is also 2, like the default model.

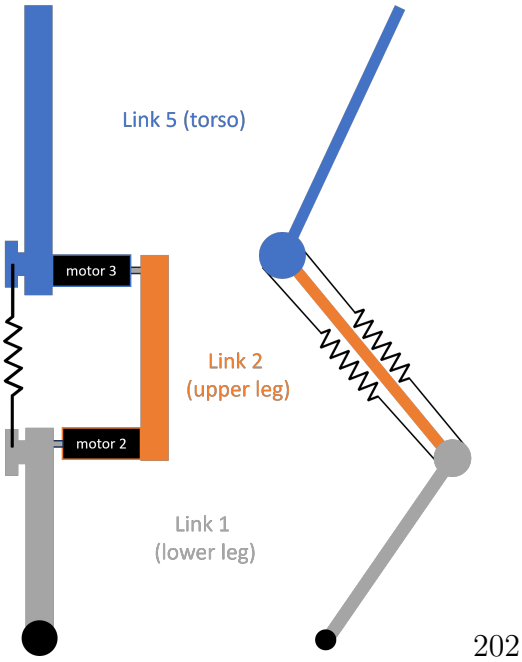


Figure 4.4: Biarticular springs are used in "parallel" configuration

4.2 Optimization

In this section, optimization setup for comparing the walking performances of the nominal and BA model will be described. Walking performance will be compared with regards to three criteria: walking efficiency, walking speed and minimum input torque requirements. The results will be presented in Section 4.4.

Optimal trajectories for the nominal and BA models will be found using direct collocation methods [9] and these results will be used to compare their performances. Direct collocation methods can be used to solve trajectory optimization problems. These methods discretize the continuous time problem and turns it into a form which can be solved by commercial nonlinear programming solvers. In this chapter, OpenOCL [10] was used to solve the trajectory optimization problem.

Optimization problem can be formulated as

$$\begin{aligned} \min_{\boldsymbol{x},\boldsymbol{u},\boldsymbol{p},T} \quad & \int_0^T J(\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{p})dt \\ \text{s.t.} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{p}) \\ & \boldsymbol{r}_k(\boldsymbol{x}_(\mu_k),\boldsymbol{p}) \leq 0, \end{aligned} \tag{4.14}$$

where $t \in [0,T]$ is the time, $\boldsymbol{x}(t)$ is the state trajectory, $\boldsymbol{u}$ are the inputs, $\boldsymbol{p}$ are the parameters, $J(\boldsymbol{x},\boldsymbol{p})$ is the path cost function, $\boldsymbol{f}(\boldsymbol{x},\boldsymbol{p})$ is the system dynamics function (differential equation) and $\boldsymbol{r}_k(\boldsymbol{x},\boldsymbol{p})$ is the grid-constraints function and $k \in \{1,...,N\}$ are the grid points (collocation points).

We used similar constraints as [3] where a similar optimization study was conducted where a comparison was made between the nominal model and a model that uses series elastic actuation (SEA). The constraints for the optimization are set as follows:

- Constraining the relative knee joint angles so that they don't bend backwards and gaits are human-like:

$$\begin{array}{c} 95^\circ < \theta_2 - \theta_1 < 112.5^\circ \\ 270^\circ < \theta_4 - \theta_3 < 360^\circ \end{array} \tag{4.15}$$

- Upper body must remain straight:

$$80^\circ < \theta_5 < 90^\circ \tag{4.16}$$

- The angular velocity of motors must not be over the desired limit:

$$|\dot{\theta}_i| < 10 \ \ [\text{rad}/s], i \in \{1, 2, 3, 4, 5\} \tag{4.17}$$

- Zero moment point should remain inside a 35 cm "foot" during the walking:

$$-0.1\ [m] < x_{\text{zmp}} < 0.25\ [m] \tag{4.18}$$

- The step length of the robot is set to be

$$0.25\ [m] : x_{hip}(T) - x_{hip}(0) = 0.25\ [m] \tag{4.19}$$

- Swing foot must start on the ground, touch the ground at the end time and it should be above the ground in the intermediate grid points ($1 < k < N$):

$$y_{sw}(0) = 0\ [m] // y_{sw}(T) = 0\ [m] \tag{4.20}$$

- In order to have a periodic trajectory, states after the collision must be the same as the initial states:

$$\boldsymbol{x}(0) = f_H(\boldsymbol{x}(T)) \tag{4.21}$$

- Mechanical parameters of the biarticular springs were constrained as:

$$\begin{array}{c} 0.01 < r < 5 \\ 0\ [Nm] < \bar{\kappa} < 2000\ [Nm]. \end{array} \tag{4.22}$$

- To prevent the swing foot from dragging on the ground, a virtual obstacle is placed (Figure 4.5):

$$\left(\frac{x_{sw} - d_{obs}}{w_{obs}}\right)^2 + \left(\frac{y_{sw}}{h_{obs}}\right)^2 \geq 1, \tag{4.23}$$

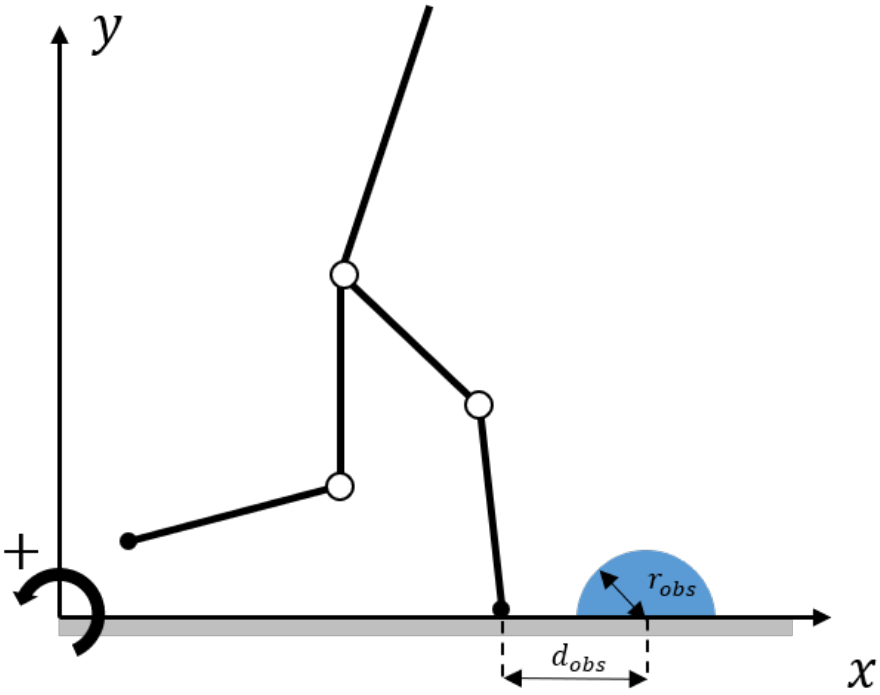


Figure 4.5: Virtual obstacle to be avoided during optimization

where x_{sw} and y_{sw} are horizontal and vertical positions of the swing foot, $d_{\text{obs}} = 0\ [m]$ is the horizontal position of the elliptic obstacle (from the stance foot), $w_{\text{obs}} = 0.1\ [m]$ and $h_{\text{obs}} = 0.02\ [m]$ are width and height of the ellipse.

- Swing foot velocity should always be positive in the intermediate grid points. (This is to avoid swinging of the foot backwards before swinging it forward to make gaits more natural. Sometimes, optimal solutions did this and it looks like this kind of swing requires less max. torque)

Number of the grid points was chosen as $N = 12$ and degree of interpolating polynomial as $d = 3$ for the optimization. For the nominal model, variables to be optimized were $\boldsymbol{x}$, $\dot{\boldsymbol{x}}$, T and $\boldsymbol{u}$. As for the BA model, $\boldsymbol{x}$, $\dot{\boldsymbol{x}}$, T, $\boldsymbol{u}$ with the addition of r and $\bar{\kappa}$ are the optimization variables. In the following subsections, criterion functions and methodology of optimization will be explained for minimizing the specific resistance (SR), maximizing the average velocity, and finding minimum input requirements.

4.2.1 Minimizing the specific resistance (SR)

First performance criteria was the walking efficiency and we have chosen the specific resistance (SR) as the cost function. SR can be calculated as:

$$SR := \frac{p}{Mgv},\ p = \frac{1}{T}\int_0^T \sum_{i=1}^{5} |u_i \omega_i| dx, \tag{4.24}$$

T [secs] is the end time of one step, M [kg] is the total weight, g $[m/s^2]$ is the gravitational term, v $[m/s]$ is the average speed, p $[J/s]$ is average input energy, u_i $[Nm]$ is the input torque and $\omega_i = \dot{\theta}_i$ $[rad/s]$ is the joint angular velocity. Robot's walking becomes more efficient as the SR value becomes smaller.

4.2.2 Maximizing the average velocity

Second performance criteria is how fast the model can move under the same constraints, i.e. the average velocity of one step. For this criterion, we've added an additional constraint $|u_i| < 50$ $[Nm], i \in \{1, ..., 5\}$, which is a constraint on the input torques.

4.2.3 Finding minimum input requirements

A model having less minimum torque requirements is desirable since it would allow the use of cheaper and smaller motors. To find the minimum torque requirements, an additional constraint was added (one step should be completed in exactly 0.8 seconds: $T = 0.8$ [sec]) so that all the resulting trajectories have the same average velocity. Then, we begin by adding a torque constraint ($u_i < 50$ $[Nm]$) and this limit was decreased by 0.1 [Nm] increments until it became unfeasible. In this study, a torque limit was considered unfeasible if OpenOCL was not able to find a feasible trajectory within 30,000 iterations. This can mean that there is no feasible solutions or the solver fails to find one in a reasonable computational time but both can be considered as unfeasible in a practical sense.

4.3 Control

In this section, a controller will be introduced which can be used to achieve stable walking gaits with a bipedal robot that has the mentioned biarticular muscles. We will propose a feedback+feedforward control to track the optimal joint trajectories obtained in Section 4.2. Feedback part of the controller will make sure that errors in the joint trajectories are compensated and the feedforward part will allow us to use the optimal input torques that were obtained during the optimization. A block diagram of this controller can be seen in Figure 4.6.

The feedforward part of the controller is given by:

$$\boldsymbol{u}_{\text{ff}}(t) = \left[u_1^{\text{opt}}(t),\ u_2^{\text{opt}}(t),\ u_3^{\text{opt}}(t),\ u_4^{\text{opt}}(t),\ u_5^{\text{opt}}(t)\right]^T, \tag{4.25}$$

where u_i^{opt}, $i \in \{1, 2, 3, 4, 5\}$ are the optimal input values for the reference trajectory. The feedback part is:

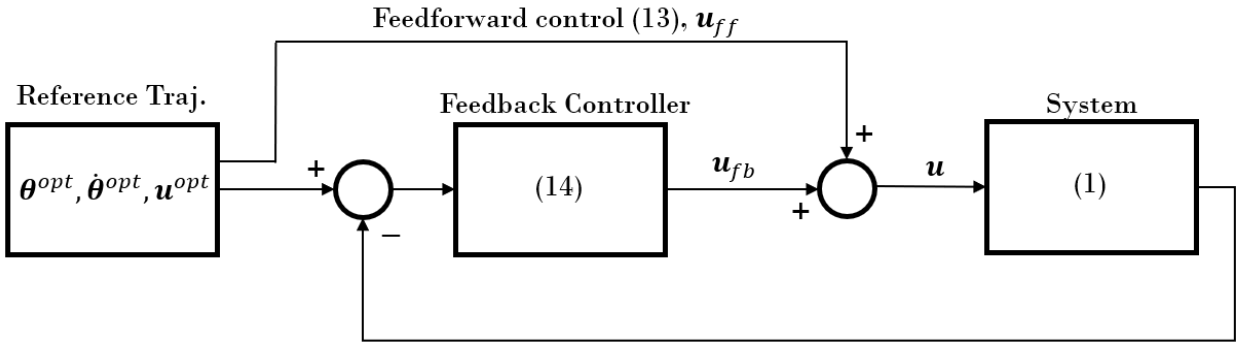


Figure 4.6: Block diagram of the controller

$$\boldsymbol{u}_{\text{fb}}(t) = \begin{bmatrix} K_{P,1}(\theta_1^{\text{opt}}(t) - \theta_1) + K_{D,1}(\dot{\theta}_1^{\text{opt}}(t) - \dot{\theta}_1) \\ K_{P,2}(\theta_2^{\text{opt}}(t) - \theta_2) + K_{D,2}(\dot{\theta}_2^{\text{opt}}(t) - \dot{\theta}_2) \\ K_{P,3}(\theta_3^{\text{opt}}(t) - \theta_3) + K_{D,3}(\dot{\theta}_3^{\text{opt}}(t) - \dot{\theta}_3) \\ K_{P,4}(\theta_4^{\text{opt}}(t) - \theta_4) + K_{D,4}(\dot{\theta}_4^{\text{opt}}(t) - \dot{\theta}_4) \\ K_{P,5}(\theta_5^{\text{opt}}(t) - \theta_5) + K_{D,5}(\dot{\theta}_5^{\text{opt}}(t) - \dot{\theta}_5) \end{bmatrix}, \tag{4.26}$$

where θ_i^{opt} and $\dot{\theta}_i^{\text{opt}}$ are the reference joint angles and velocities of the reference trajectory (the method for obtaining these trajectories was described in Section 4.2) and t is the time from the beginning of the current step (time since the last impact). Since the direct collocation has only 12 grid points, we linearly interpolated the reference trajectory to obtain the intermediate values. If the time of one step becomes longer than the reference trajectory, final value of the reference was used. The total control values is obtained as:

$$\boldsymbol{u} = \boldsymbol{u}_{\text{ff}} + \boldsymbol{S}^{-1}\boldsymbol{u}_{\text{fb}}. \tag{4.27}$$

4.4 Results and Discussion

In Section 4.2, optimization study where the walking performance of the nominal model was compared with the performance of a model that has passive biarticular muscles was described. Results of the described optimization with respect to three different criteria are presented in Table 4.1. We used default metrics of success for the optimization with OpenOCL which can be seen in its documentation [10]. In this table, resulting average velocities, specific resistance values, max torques, and mechanical parameters for the BA springs can be seen.

Table 4.1 shows that with the use of biarticular springs, walking efficiency can be increased significantly under the same constraints. For the optimized trajectories, SR value of the nominal model is 0.05 and it is 0.02 for the BA model. It is also interesting to note that the BA model can walk more efficiently while also achieving a faster average velocity than the nominal model.

Table 4.1: Optimization Results

		Nom. Model	BA Model
Min. SR	Avg. Vel. [m/s]	0.25	0.30
	SR	**0.05**	**0.02**
	Max. Torque [Nm]	34.54	37.86
	r	-	1.65
	$\bar{\kappa}$	-	63.87
Max. Avg. Vel.	**Avg. Vel. [m/s]**	**0.95**	**1.35**
	SR	0.35	0.43
	Max. Torque [Nm]	50	50
	r	-	3.93
	$\bar{\kappa}$	-	75.48
Min. Input Req.	Avg. Vel. [m/s]	0.31	0.31
	SR	0.13	0.07
	Input Req. [Nm]	**13.9**	**4.6**
	r	-	2.13
	$\bar{\kappa}$	-	32.85

Use of passive biarticular muscles also helped the bipedal model achieve faster walking average velocities for the second optimization criterion. The trajectory that the solver found for the nominal model had an average velocity of 0.95 [m/s] and with the addition of biarticular springs, this value was increased to 1.35 [m/s] under the same maximum motor torques constraints. However, reaching a larger average velocity value for the BA model came at the price of a larger SR.

For the final criteria, BA model once again outperformed the nominal model significantly. The minimum torque limit that the solver was able to find feasible solutions for the nominal model was 13.9 [Nm] and for the BA model, this was much lower at 4.6 [Nm]. Both trajectories had the same average velocities as per the added constraint. SR value for the BA model is again lower for this criterion.

If we take a look at the resulting biarticular spring parameters, it can be seen that the stiffness value for finding the minimum input torque requirement was the smallest. We believe that this is because larger BA stiffness values require larger inputs to control, as can be seen in the resulting values for minimizing SR criterion. In the first criteria, even tough the SR value is smaller for the BA model, the maximum torque value is larger so that torques generated by the BA springs can be controlled.

Then, we conducted a simulation study to test the performance of the controller that was proposed in Section 4.3. For the reference trajectory, we chose the resulting trajectory of optimization with minimizing SR criterion. Simulations were performed in Matlab SIMULINK environment with ode45 solver and variable step settings (absolute tolerance was set to 1e-8).

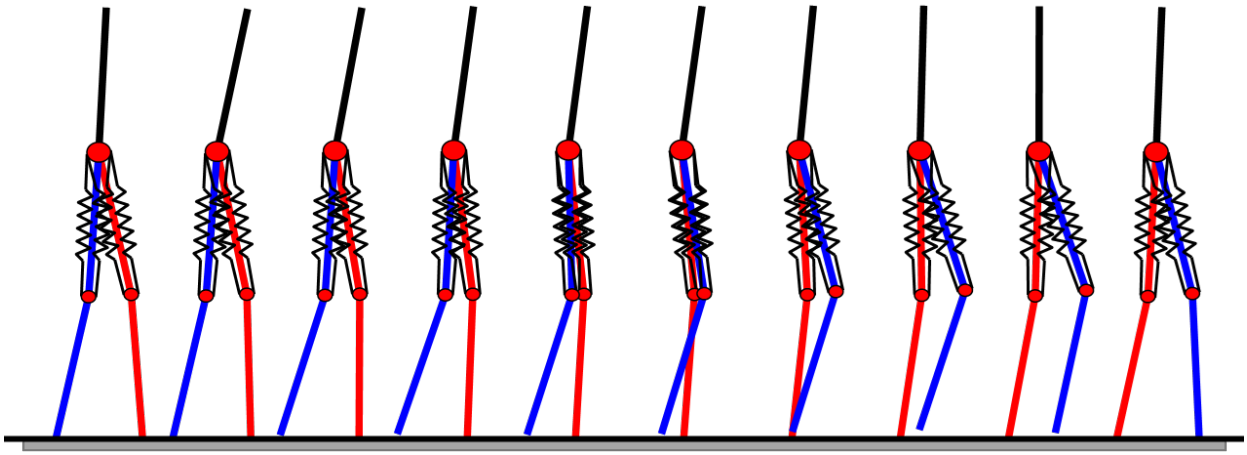


Figure 4.7: Snapshots of one step (BA Model)

Table 4.2: 5 Link Model Parameters

$l_1 = l_4 : 0.48[m]$	$l_2 = l_3 : 0.48[m]$	$l_5 : 0.48[m]$
$m_1 = m_4 : 5[kg]$	$m_2 = m_3 : 5[kg]$	$m_5 : 60[kg]$
$I_i = m_i l_i^2/12[kg \cdot m^2], i = 1, 2, 3, 4, 5$		

The mechanical parameters of the model are presented in Table 4.2. Mechanical parameters of the biarticular muscles were chosen as $r = 1.65$, $\bar{\kappa} = 63.87$ which are the optimization results for minimizing SR. $r_h = 0.02$ $[m]$ was chosen arbitrarily with regards to the overall size of the model.

We have tested two versions of the controller. In the first version, we set $K_{P,i} = 550$, $K_{D,i} = 40$, $i \in \{1, ..., 5\}$ in (4.26). This version of the controller has only two gain variables and it is easy to tune by hand. In the second version, there are ten different gain variables and we tuned them using a simple particle swarm optimization [12] algorithm with SR as the criterion function to be minimized. Resulting gain values are presented in Table 4.3.

Using the controller with two gain variables, a stable tracking can be achieved with SR = 0.04 and an average velocity of 0.29 [m/s]. This SR value is still lower than that of the nominal model's optimized trajectory with respect to SR, but not very close to the optimal SR value of the BA model. When the controller with individual gains for each joint (Table

Table 4.3: Controller Parameters

$K_{P,1} = 445.3$	$K_{D,1} = 17.7$
$K_{P,2} = 307.3$	$K_{D,2} = 34.0$
$K_{P,3} = 90.4$	$K_{D,3} = 12.0$
$K_{P,4} = 18.6$	$K_{D,4} = 1.4$
$K_{P,5} = 275.2$	$K_{D,5} = 26.6$

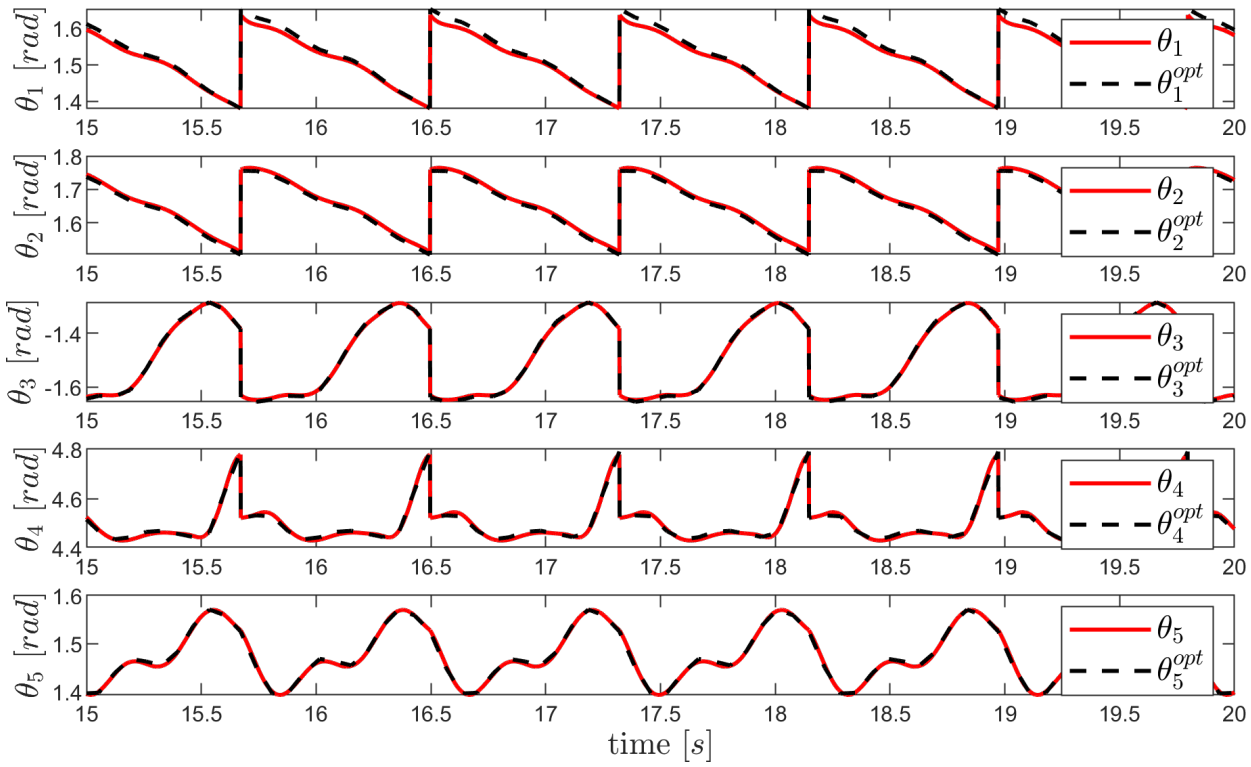


Figure 4.8: Trajectory tracking results for the joint angles (BA Model)

4.3) was used, we were able to obtain a stable walking gait with SR = 0.03 and an average velocity of 0.29 [m/s]. This SR value is much closer to the optimal value of SR = 0.02 but still slightly larger. This difference might be due to reference trajectory having only 12 grid points and the intermediate values were obtained using linear interpolation. Fitting a higher order polynomial to the reference trajectory or increasing the number of grid points can improve the tracking performance but optimizing with larger number of grid points are harder to solve.

Figure 4.7 shows snapshots of one step and Figure 4.8 shows the trajectories of each joint when the controller gain variables are chosen as in Table 4.3. It can be seen that tracking performance is quite satisfactory.

Stability of this trajectory was tested using Poincaré stability criterion. The axes of the Poincaré map were selected as the $\theta_P = \text{atan2}(\dot{y}_{\text{CoM}}, \dot{x}_{\text{CoM}})$, y_{CoM} and total energy of the 5 link model E. Poincaré section was chose as the vertical leg orientation (VLO) point. VLO is when CoM of the 5 link model is directly above the stance foot ($x_{\text{CoM}} = x_{\text{stance foot}}$). VLO was chosen as the Poincaré section because the horizontal position doesn't need to be considered at this point. Poincaré stability criterion indicates that if the return map converges to a fixed point, a hybrid system with impact effects can be considered periodic [13]. A two dimensional section of the Poincaré Map of the trajectory where the proposed controller is used can be seen in Figure 4.9 where the return map converges to a stable point.

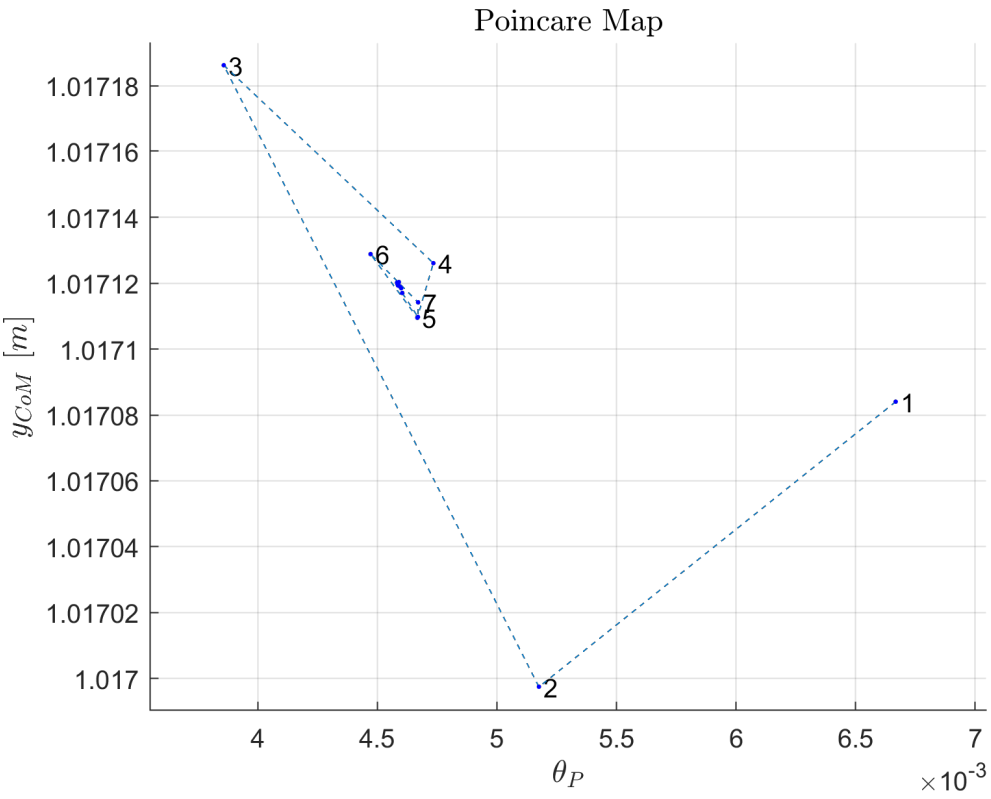


Figure 4.9: 2D section of the Poincaré Map where the numbers indicate the step numbers

4.5 Conclusion

In this chapter, we conducted an optimization study to assess the walking performance of a fully-actuated 5-link planar bipedal walker with and without biarticular springs. The goal was to compare the performance of these two models based on several criteria, including walking efficiency, walking speed, and minimum input torque requirements.

Through our optimization analysis, we observed that the model employing biarticular springs (referred to as the BA model) consistently outperformed the nominal model in all investigated criteria. Specifically, the BA model exhibited superior walking efficiency, allowing for more efficient energy utilization during locomotion. Moreover, the BA model achieved higher walking speeds compared to the nominal model, indicating enhanced overall performance.

Another noteworthy advantage of the BA model was the reduced minimum input torque requirements. By incorporating biarticular springs, the BA model could effectively distribute forces and leverage the mechanical advantages of biarticular muscles, resulting in a more compact and economical design. This aspect is particularly significant as it enables the utilization of smaller motors, leading to potential cost savings and improved efficiency in practical applications.

Subsequently, we introduced a feedback-plus-feedforward controller to achieve stable walk-

ing gaits with the bipedal robot model equipped with passive biarticular muscles (referred to as the BA model). The objective of the proposed controller was to accurately track the reference trajectories obtained in the optimization phase.

Overall, the proposed controller provided a comprehensive control strategy for achieving stable and accurate walking gaits with the bipedal robot model incorporating passive biarticular muscles. By successfully tracking the reference optimal trajectories and achieving similar SR values to the them, the controller demonstrated its efficacy in enabling the robot to achieve the desired walking behaviors and maintain stability throughout the locomotion process.

Chapter 5

Effects of Active Wobbling Mass on Biped Robot's Walking Performance in Combination with Biarticular Springs

Human walking is characterized by pronounced arm movement during walking and running motions. However, many bipedal robots and walking models ignore the arm-swing. This motion is usually symmetric and is in sync with the walking motion. Amplitude of the arm swing changes with respect to the walking speed. Amplitudes become larger as the walking speed increases [23].

Arm swing is an integral part of human bipedal gait, arising mostly from passive movements, which are stabilized by active muscle control and arm swinging during normal bipedal gait most likely serves to reduce energy expenditure [24]. In [25], researchers suppressed the arm movements of human subjects and found out that walking energy expenditure was significantly increased.

Addition of arms can be used to increase walking stability and speed but having multiple degree of freedom actuators attached to the torso might significantly increase the weight due to additional actuators and links and also it increases the complexity of the walking controller. To overcome these issues, researchers have investigated using the similarity between swinging arms in humans and the up-and-down motions of a reaction mass (wobbling mass) Figure 5.1.

Rome et al. have shown that walking performance of humans with a heavy backpack is improved by up-and-down motions of the backpack using elastic elements [26]. Tanaka et al. have shown increasing walking speed of a combined rimless wheel that can be achieved by using up-and-down motions of a wobbling mass [27]. In [23], it was numerically and mathematically shown that a controlled wobbling mass was able to increase the converged walking speed of 3-link bipedal robot model with arc-feet. We investigated the effects of the wobbling mass in combination with rotational inerters and series elastic actuators through

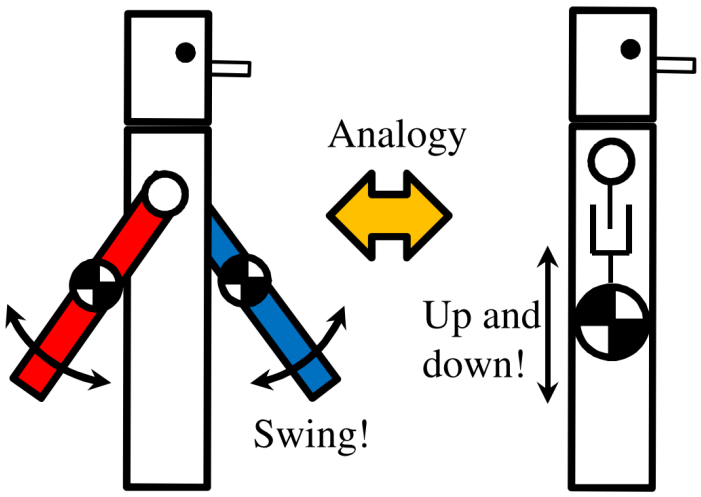


Figure 5.1: Analogy of swinging arms and wobbling mass

trajectory optimization studies and found that energy consumption can be reduced by 30% [28].

In this chapter, we will investigate the effects of adding a controlled wobbling mass to the torso of a 5-link underactuated bipedal robot on walking speeds. This will be done through trajectory optimization studies. In Chapter 4, we've shown that adding biarticular springs to the bipedal walking robot can improve energy efficiency, walking speed and minimum torque requirements. In this chapter, we prose that using biarticular springs in combination with a wobbling mass could further improve the walking speeds of a biped walker.

5.1 Systems and Modeling

5.1.1 Wobbling Mass

The wobbling mass is modeled as a point mass that can move up-down inside the torso of the bipedal robot via a linear actuator (leadscrew actuator etc.). The movement of the mass in constrained within the torso. Arm swing movement can be represented with the movement of the wobbling mass a shown in Figure 5.2. We will model the input that moves the wobbling mass as force and this will be represented by the u_{wm} [N] term where "wm" subscript indicate the "wobbling mass".

5.1.2 Bipedal Robot Model

In this chapter, we will conduct the trajectory optimization studies on a 5-link underactuated bipedal robot model. This model is considered to be point-footed, i.e. there is no ankle torque. It has 2 actuators on the knees and 2 actuators on the hip where the revolute joints are positioned. Also, there are springs connected in a biarticular configuration between the torso and the lower legs same as in Chapter 4. We consider the model to be constrained to

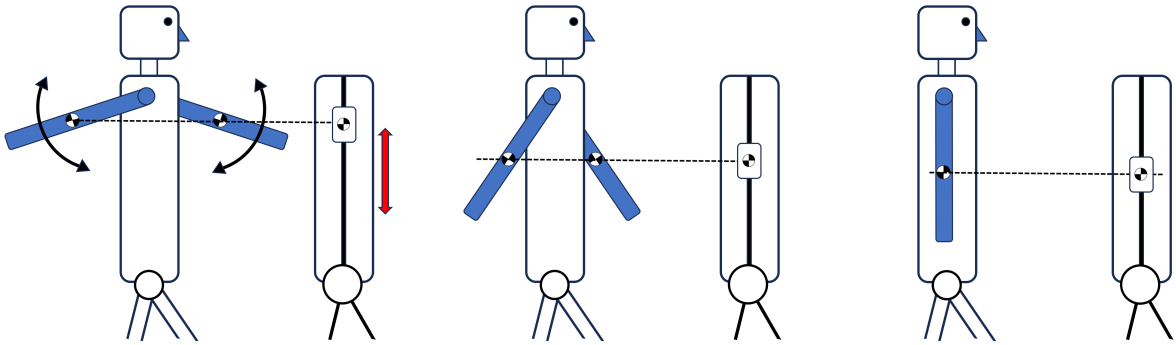


Figure 5.2: Human arm swing motions (left figures) versus the up and down motion of the wobbling mass inside the bipedal robot torso (right figures)

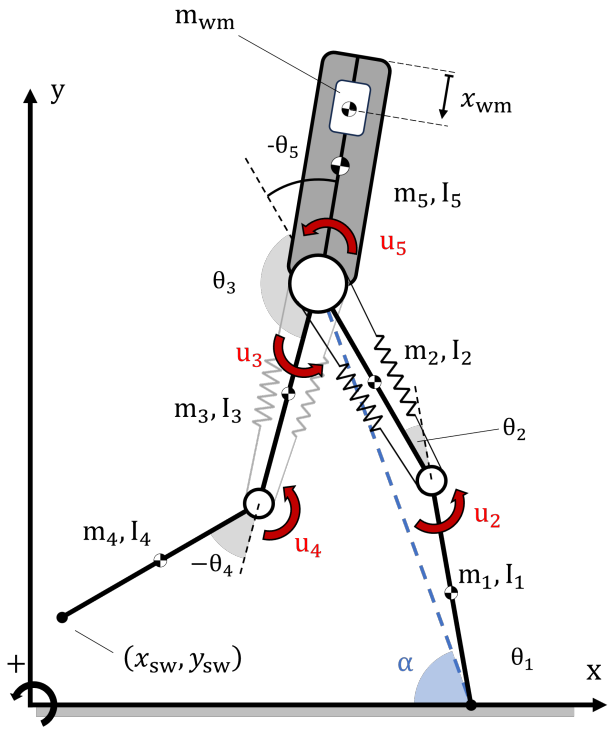


Figure 5.3: Human arm swing motions (left figures) versus the up and down motion of the wobbling mass inside the bipedal robot torso (right figures)

Model No.	Wobbling mass	Biarticular Springs
1	NO	NO
2	YES, LOCKED	NO
3	YES	NO
4	NO	YES
5	YES, LOCKED	YES
6	YES	YES

Table 5.1: Different models compared in this chapter where Model 1 is the nominal model and model 6 is the proposed model

the saggital plane.

We will compare the walking speed performance of 6 different models shown in Table 5.1. These model are the combinations of including or no includings biarticular muscles, including or not including the wobbling mass, and locking or not locking for the cases we include the wobbling mass.

Equation of motion of this model with biarticular springs and wobbling mass can be written as:

$$\mathbf{M}_{\text{wm}}(\mathbf{q}_{\text{wm}})\ddot{\mathbf{q}}_{\text{wm}} + \mathbf{H}_{\text{wm}}(\mathbf{q}_{\text{wm}}, \dot{\mathbf{q}}_{\text{wm}}) = \mathbf{S}_{\text{wm}}\mathbf{u}_{\text{wm}} + \boldsymbol{\tau}_{\text{wm}}(\mathbf{q}_{\text{wm}}), \tag{5.1}$$

where

$$\mathbf{q}_{\text{wm}} = [\mathbf{q}^{\text{T}}, x_{\text{wm}}]^{\text{T}} \in \mathbb{R}^6, \tag{5.2}$$

are the generalized coordinates and

$$\mathbf{q} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^{\text{T}}, \tag{5.3}$$

$\mathbf{M}_{\text{wm}}(\mathbf{q}_{\text{wm}}) \in \mathbb{R}^{6\times6}$ is the inertia matrix, $\mathbf{H}_{\text{wm}}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^6$ is the Coriolis, centrifugal and gravitational terms vector, $\mathbf{S}_{\text{wm}} \in \mathbb{R}^{6\times5}$ is the distribution matrix of the inputs,

$$\mathbf{u}_{\text{wm}} = [\mathbf{u}^{\text{T}}, u_{\text{wm}}]^{\text{T}} \in \mathbb{R}^5, \tag{5.4}$$

where

$$\mathbf{u} = [u_2, u_3, u_4, u_5]^{\text{T}}, \tag{5.5}$$

are the inputs. $u_{2,3,4,5}$ [Nm] are torques and u_{wm} [N] is a force. $\boldsymbol{\tau}_{\text{wm}} \in \mathbb{R}^6$ represents the torques generated by the biarticular springs. $\boldsymbol{\tau}_{\text{wm}}$ can be expanded as:

$$\boldsymbol{\tau}_{\text{wm}} = \mathbf{B}(\boldsymbol{\tau}_{\text{st}} + \boldsymbol{\tau}_{\text{sw}}), \tag{5.6}$$

where subscripts “st" and "sw" respectively represent the "stance leg" which is the leg that is in contact with the ground and the other leg called the "swing leg". $\mathbf{B}$ is that matrix that maps the biarticular spring torques to the model with wobbling mass coordinates:

$$\mathbf{B} = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \\ 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}^{6\times 5}. \tag{5.7}$$

We calculate the biarticular spring torques in the same manner as [29] where the partial derivative of the potential energy stored in the springs is taken with respect to the generalized coordinates, resulting in:

$$\boldsymbol{\tau}_{\mathrm{st}} = \begin{bmatrix} 0 \\ -\kappa\, r_{\mathrm{k}} \Delta l_{\mathrm{st}} \\ 0 \\ 0 \\ -\kappa\, r_{\mathrm{h}} \Delta l_{\mathrm{st}} \end{bmatrix},\ \boldsymbol{\tau}_{\mathrm{sw}} = \begin{bmatrix} 0 \\ 0 \\ \kappa\, r_{\mathrm{h}} \Delta l_{\mathrm{sw}} \\ \kappa\, r_{\mathrm{k}} \Delta l_{\mathrm{sw}} \\ -\kappa\, r_{\mathrm{h}} \Delta l_{\mathrm{sw}} \end{bmatrix}, \tag{5.8}$$

where κ [N/m] is the biarticular spring stiffness, r_{h} [m] and r_{k} [m] are the lever arm lengths with subscripts "h" and "k" referring to "hip" and "knee".

$$\Delta l_n = r_h(\varphi_h^n - \varphi_{h0}) - r_k(\varphi_k^n - \varphi_{k0}),\ n \in \{\mathrm{sw}, \mathrm{st}\}, \tag{5.9}$$

are the deflection of the respective spring where.

$$\begin{aligned} \varphi_h^{\mathrm{sw}} &= \theta_5 - \theta_3 \\ \varphi_k^{\mathrm{sw}} &= \theta_4 \\ \varphi_h^{\mathrm{st}} &= \theta_5 - \pi \\ \varphi_k^{\mathrm{st}} &= \pi - \theta_2 \end{aligned}. \tag{5.10}$$

We introduce the lever arm ratio $r = r_h/r_k$, and new spring constant term $\bar{\kappa} = \kappa\, r_{\mathrm{k}}^2$ and deflection

$$\bar{\Delta l_n} = \frac{\Delta l_n}{r_{\mathrm{k}}}, \tag{5.11}$$

so that we can search for r and $\bar{\kappa}$ [Nm] instead of the r_{h}, r_{k} and κ [N/m] during trajectory optimization. Equation (5.8) can be rewritten using these new terms.

Equation (5.1) models the single stance phase (when one foot is on the ground and the other is doing the swinging motion). When the swing foot contacts the ground (touch-down), an impact occurs and model goes to a double stance phase where both foot are on the ground. In this chapter, we assume an instantaneous double stance phase i.e. swing leg and stance leg switch instantaneously at the moment of the impact and after the impact, swing leg lifts up from the ground without interaction (lift-off). The reset map is given by:

$$\boldsymbol{x}^+ = f_H(\boldsymbol{x}^-). \tag{5.12}$$

where $\boldsymbol{x} \in \{\boldsymbol{x}_{\mathrm{nom}}, \boldsymbol{x}_{\mathrm{wm}}\}$.

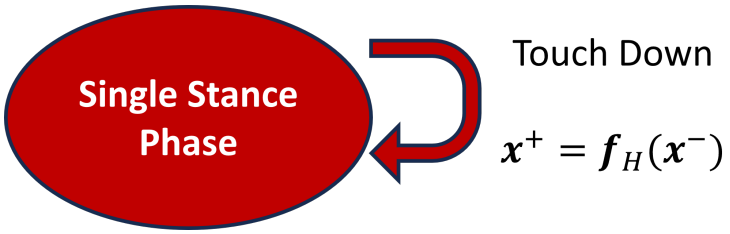


Figure 5.4: State transition

Table 5.2: 5-link bipedal robot model with a wobbling mass mechanical parameters

$l_1 = l_4 : 0.48$ [m]	$l_2 = l_3 : 0.48$ [m]	$l_5 : 0.48$ [m]
$m_1 = m_4 : 5$ [kg]	$m_2 = m_3 : 5$ [kg]	$m_5 : 60$ [kg]
$m_{\text{wm}} = 5$ [kg]		
$I_i = m_i l_i^2/12$ [kg $\cdot$ m^2], $i = 1, 2, 3, 4, 5$		

$$\begin{gathered} \boldsymbol{x}_{\text{nom}} = [\mathbf{q}^{\text{T}}, \dot{\mathbf{q}}^{\text{T}}]^{\text{T}} \\ \text{or} \\ \boldsymbol{x}_{\text{wm}} = [\mathbf{q}_{\text{wm}}^{\text{T}}, \dot{\mathbf{q}}_{\text{wm}}^{\text{T}}]^{\text{T}} \end{gathered} \tag{5.13}$$

depending on the model. During this event, position of the robot remains the same, only the swing leg and stance leg are swapped but velocities change discontinuously [29]. State transition diagram is shown in Figure 5.4.

Model whose dynamics described by Equation (5.1) corresponds to Model 6. Models without biarticular muscles can be described by setting the spring stiffness value to zero ($\kappa = 0$).

Model 1's (nominal model) dynamics are described by:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{S}\mathbf{u}. \tag{5.14}$$

For the locked wobbling mass models, we use the nominal modals equation of motion (Equation 5.14) but add the weight of the wobbling mass to the torso's weight:

$$m_{5,\text{wml}} = m_5 + m_{\text{wm}}, \tag{5.15}$$

where subscript "wml" means "wobbling mass locked". We set the parameters for this model as shown in Table 5.2. Biarticular spring stiffness κ [N/m] and lever arm ration r terms will be obtained via the simultaneous trajectory and parameter optimizations.

5.2 Simultaneous Trajectory and Parameter Opimization

In this section, we will describe the optimization setup that was used in obtaining the reference trajectories for regular walking and stepping-down motions using direct collocation method [9]. These methods turn the continuous time problem into discrete one which then can be handled by nonlinear programming solvers. In this chapter, OpenOCL [10] was used to solve the trajectory optimization problem.

For the nominal model or the wobbling mass locked model, optimization problem can be formulated as:

$$\begin{aligned} \min_{\boldsymbol{x},\mathbf{u}_{\mathrm{wm}},\boldsymbol{p},T} \quad & \int_0^T L(t,\boldsymbol{x}(t),\mathbf{u}_{\mathrm{wm}}(t),\boldsymbol{p})dt \\ \text{s.t.} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t),\mathbf{u}(t),\boldsymbol{p}) \\ & \boldsymbol{r}(\boldsymbol{x},t,\boldsymbol{p}) \leq 0, \end{aligned} \tag{5.16}$$

where $t \in [0,T]$ is the time, $\boldsymbol{x}(t)$ is the state trajectory, $\mathbf{u}(t)$ are the inputs, $\boldsymbol{p}$ are the parameters, $L(\boldsymbol{x}(t),\mathbf{u}_{\mathrm{wm}}(t),\boldsymbol{p})$ is the path cost function, $\boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{p})$ is the system dynamics function (differential equation) and $\boldsymbol{r}(\boldsymbol{x},t,\boldsymbol{p})$ are the constraint functions.

The dynamic constraints and the inequality constraints are realized on grid points (collocation points). Number of the grid points was chosen as $N = 12$ and degree of interpolating polynomial as $d = 3$ for the optimization.

The cost function was set to:

$$L(t) = T. \tag{5.17}$$

in order to maximize the average walking speed under actuation torque constraints.

We determined a set of constrains so that resulting trajectory is a human-like walking gait:

- Constraining the relative knee joint angles to achieve human-like gaits:

$$\begin{aligned} 5^\circ < \theta_2 < 22.5^\circ \\ 270^\circ < \theta_4 < 345^\circ \end{aligned} \tag{5.18}$$

- Upper body must remain straight:

$$80^\circ < \theta_5 < 90^\circ \tag{5.19}$$

- The angular velocity of motors must not be over the desired limit:

$$|\dot{\theta}_i| < 10 \ \ [\mathrm{rad}/s], i \in \{1,2,3,4,5\} \tag{5.20}$$

- Center of mass of the robot should always be moving with positive velocity in the x direction:

$$\dot{x}_{\text{CoM}}(t) > 0 \text{ [m/s]} \tag{5.21}$$

- Setting a lower bound for virtual stance leg angle $\boldsymbol{\alpha}$'s velocity to keep it monotonically increasing (we set the lower bound to 0.3 [rad/s] rather than to 0 [rad/s] in order to have some safety margin in the case of blind walking in rough terrain):

$$\dot{\boldsymbol{\alpha}} > 0.3 \text{ [rad/s]} \tag{5.22}$$

- The step length of the robot is set to be 0.25 [m]
- Swing foot related constraints: $y_{\text{sw}}(0) = y_{\text{sw}}(T) = 0$ $[m]$, $y_{\text{sw}}(0 < t < T) > 0$ [m], $\dot{y}_{\text{sw}}(T) < -0.2$ [m/s], $\dot{x}_{\text{sw}}(T) < 0$ [m/s], $\dot{x}_{\text{sw}}(t < T) > 0$ [m/s]
- The trajectory must be periodic: $\boldsymbol{x}(0) = f_H(\boldsymbol{x}(T))$ (f_H is the reset map in Equation 5.12)
- Mechanical parameters of the biarticular springs were constrained as:

$$\begin{gathered} 0.01 \leq r \leq 5 \\ 0 \ [Nm] \leq \bar{\kappa} \leq 2000 \ [Nm] \end{gathered} \tag{5.23}$$

- Swing foot must avoid a virtual elliptic obstacle (prevents foot dragging and keeps to swing leg from contacting the ground early on rough terrains):

$$\left(\frac{x_{\text{sw}}(t) - d_{\text{obs}}}{w_{\text{obs}}}\right)^2 + \left(\frac{y_{\text{sw}}(t)}{h_{\text{obs}}}\right)^2 \geq 1, \tag{5.24}$$

  where x_{sw} and y_{sw} are horizontal and vertical positions of the swing foot, $d_{\text{obs}} = 0$ $[m]$ is the horizontal position of the elliptic obstacle (from the stance foot), $w_{\text{obs}} = 0.2$ $[m]$ and $h_{\text{obs}} = 0.05$ $[m]$ are the width and height of the ellipse.
- Wobbling mass must stay withing the torso:

$$0 \leq x_{\text{wm}} \leq l_5 \text{ [m]} \tag{5.25}$$

- Constraining the input torques:

$$u_i < 50 \text{ [Nm]}, \ i \in \{2, 3, 4, 5\} \tag{5.26}$$

5.3 Results and Discussion

The trajectory optimization results are shown in Table 5.3 where specific resistance (SR) is a walking efficiency term that is defined in more detail in [38]. Energy efficiency increases

Model No.	WM	BA	SR	**Avg.Vel. [m/s]**
1	NO	NO	0.62	**0.71**
2	LOCKED	NO	0.44	**0.69**
3	YES	NO	0.56	**0.74**
4	NO	YES	0.41	**1.03**
5	LOCKED	YES	0.38	**0.95**
6	YES	YES	0.47	**1.19**

Table 5.3: Trajectory optimization results for maximizing average velocity

as SR becomes smaller. We can see that model that was able to move the fastest under the same constrains was the one that utilizes the active wobbling mass and biarticular springs together. This might indicate that arm swing plays an important role in achieving faster walking speeds in human gait.

The fastest trajectory that moves the nominal model (nominal model doesn't have biarticular springs or a wobbling mass, Model #1) has converged at 0.71 [m/s] average walking velocity. When we add a locked wobbling mass to the nominal model (Model #2), walking speed decreased by 3.4 %. This is reasonable since the locked wobbling mass just increases the total weight of the system which which makes it more difficult on the actuators that have a maximum torque output of 50 [Nm].

When we add a wobbling mass that can move up and down to the nominal model, walking speed increased to 0.74 [m/s], which is a 3.8 % increase. This shows that wobbling mass can increase the walking speed by itself. However, this increase is much less than that was found in Hanazawa et al. [23]. They were able to increase the walking speed of a 3-link model from 0.59 [m/s] to 0.73 [m/s] which is a 23.7 % increase. However, they used a 2.5 [kg] wobbling mass for a 15 [kg] robot whereas we used a 5 [kg] wobbling mass for a 80 [kg] robot. This might indicate that wobbling mass is more effective in increasing the walking speed as its ratio to the robot mass is increases.

By adding only biarticular springs to the nominal model, walking speed increased by a 44.7 % to 1.03 [m/s]. Table 5.4 shows the biarticular spring parameters that the solver outputs for corresponding trajectories. Again, just adding a locked wobbling mass reduced the converged average velocity, however it is still faster than the nominal model. When we use biarticular springs in combination with wobbling mass (Model #6), the top speed was achieved at 1.19 [m/s]. This is a 66.9 % increase compared to the nominal model.

Now, let's compare the performance between the model that have biarticular springs. By adding wobbling mass, average walking velocity increased from 1.03 [m/s] for Model #3 to 1.19 [m/s] for Model #6. This is a 15.3 % increase. This increase was just 3.8 % for the models without biarticular springs. It seems that wobbling mass is more effective in increasing the walking speeds for systems that have compliant elements and/or biarticular springs.

Model No.	$\bar{\kappa}$ [Nm]	r
4	247.96	2.07
5	204.13	2.02
6	332.02	2.17

Table 5.4: Biarticular muscle parameteres obtained from simulatenous trajectory and parameter optimizations

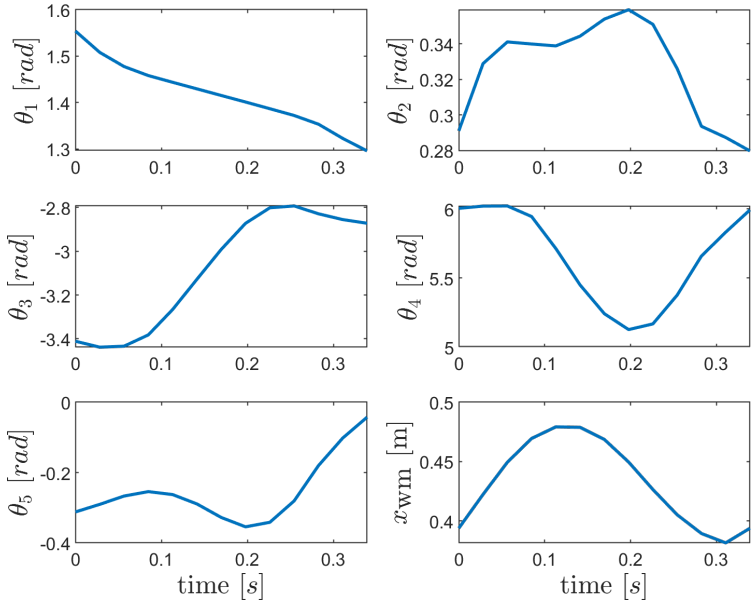


Figure 5.5: Resulting trajectories for Model #3

5.3.1 Resulting Trajectories

Figure 5.5 shows resulting trajectories resulting from the direct collocation optimization. Figure 5.6 shows the inputs used to achieve these trajectories and Figure 5.7 shows some snap-shots from the gait.

Figure 5.8 shows resulting trajectories resulting from the direct collocation optimization. Figure 5.9 shows the inputs used to achieve these trajectories and Figure 5.10 shows some snap-shots from the gait.

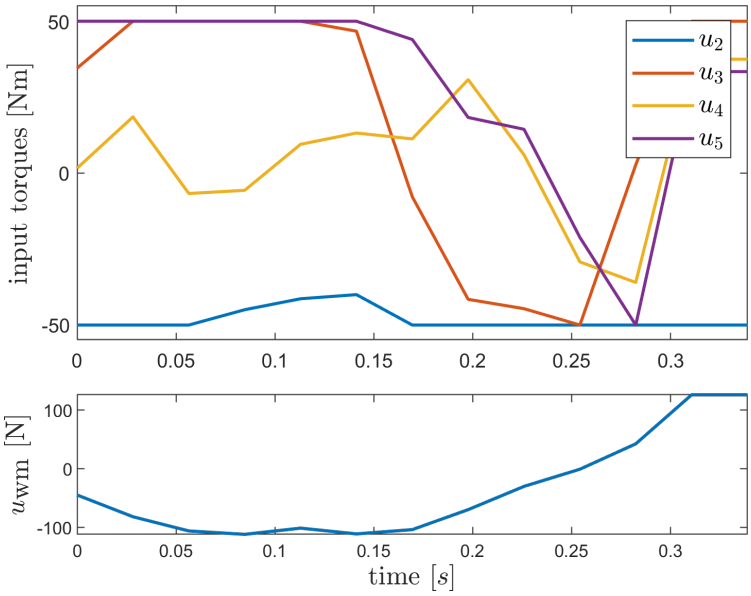


Figure 5.6: Resulting inputs for Model #3

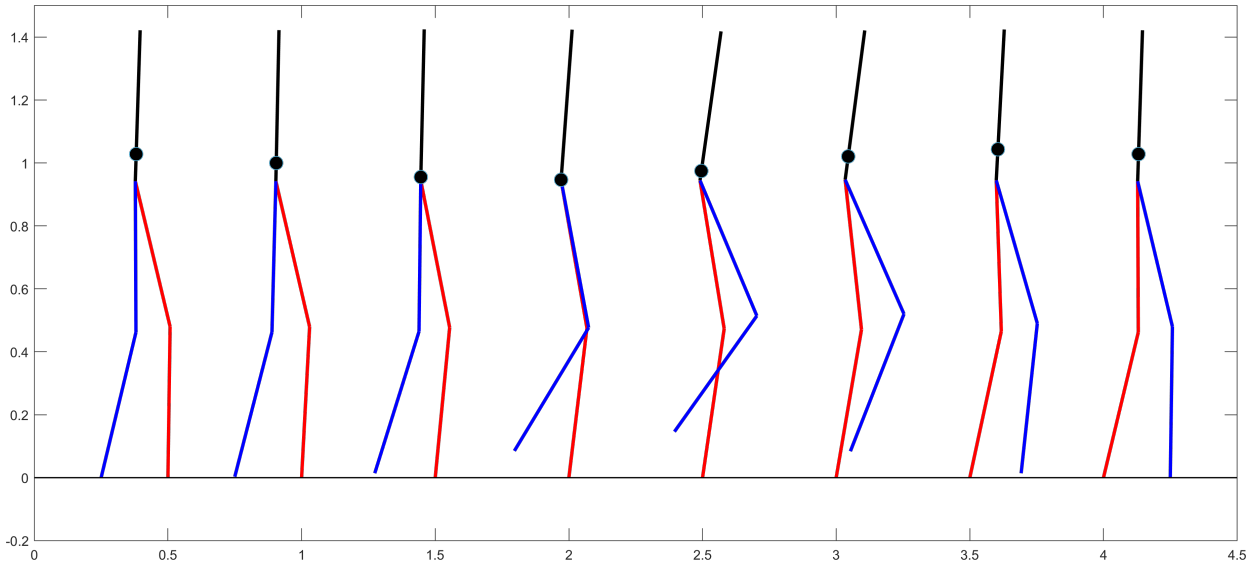


Figure 5.7: Resulting walking snap-shots for Model #3

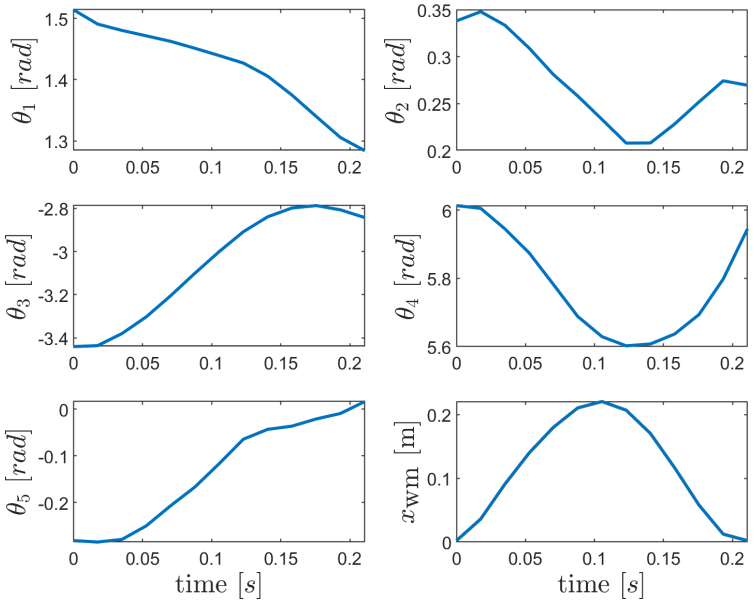


Figure 5.8: Resulting trajectories for Model #6

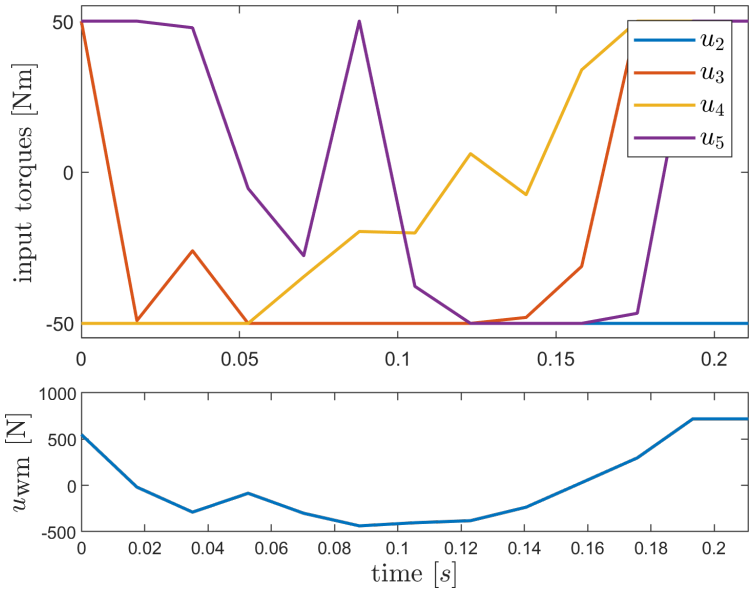


Figure 5.9: Resulting inputs for Model #6

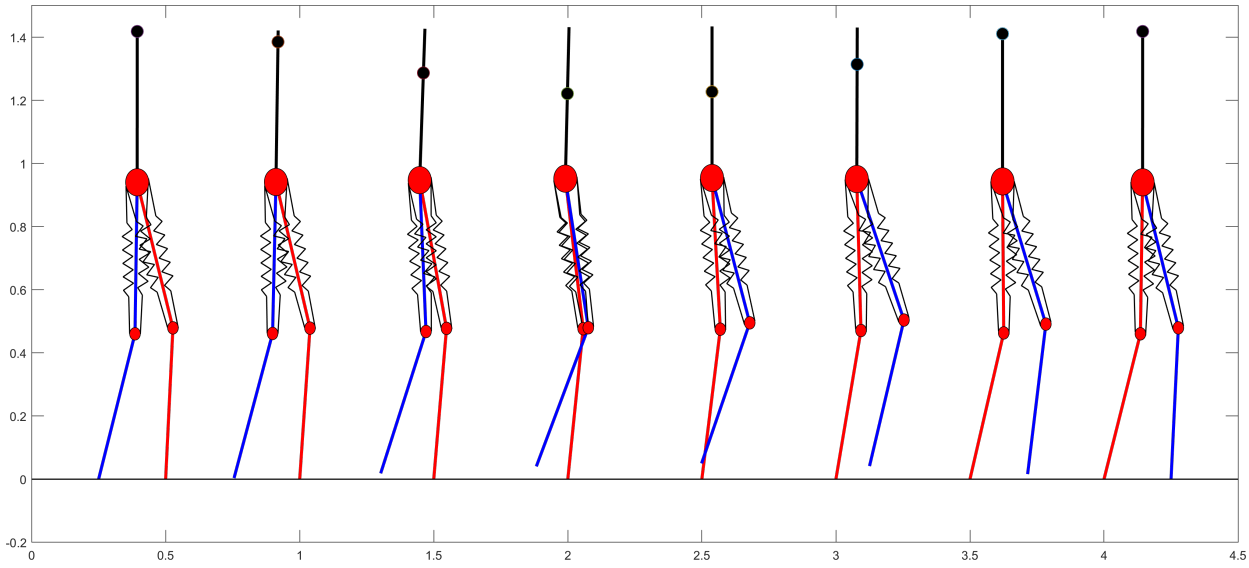


Figure 5.10: Resulting walking snap-shots for Model #6

Chapter 6

Terrain-Blind Humanoid Walking on Rough Terrain with Trajectory Optimization and Biarticular Springs

Human gait is efficient and very robust but bipedal robots are yet to mimic this success. One reason why bipedal robots have a hard time operating in real-world environments is because of rough terrains such as gravel roads, farming fields, forests etc. This is why in this chapter, we tried to improve the robustness of a 5-link underactuated bipedal robot when it moves in unknown (blind walking) rough terrain via the addition of passive biarticular muscles along with our proposed controller based on optimized trajectories.

Trajectory optimization techniques are commonly used for bipedal robots because they can provide optimal or locally optimal gaits to the chosen cost functions. They are increasingly gaining interest due to the improvements in computational power and commercially available solvers that can handle constrained non-linear problems. In [30], non-linear programming with basic splines was utilized to solve the trajectory optimization problem of a biped robot with series elastic actuators by assuming a fixed-base model. In [31], a library of optimal trajectories were used to achieve speed tracking in the 3D actuated robot ATRIAS. Optimization code used to generate the trajectories that make up the gait library is explained in [32] where they used direct transcription methods to utilize nonlinear programming solvers.

In [33], direct collocation methods were used to find an optimal trajectory for a template model called spring-loaded inverted pendulum model with swing legs. Then the reference trajectories from the resulting template model was used to control a 5-link fully actuated robot. Studies mentioned above show the effectiveness of trajectory optimization methods in achieving bipedal gait but they all assume the walking occurs on a flat surface and the robustness of these methods are questionable.

One important factor of robustness for biped robots is their ability to walk over uneven or rough terrain. There are many proposed controllers that tackle the walking on known uneven terrain issue but the ability to overcome changes in ground height without perceiving the environment is still an open research problem. If a robot is able to walk on rough

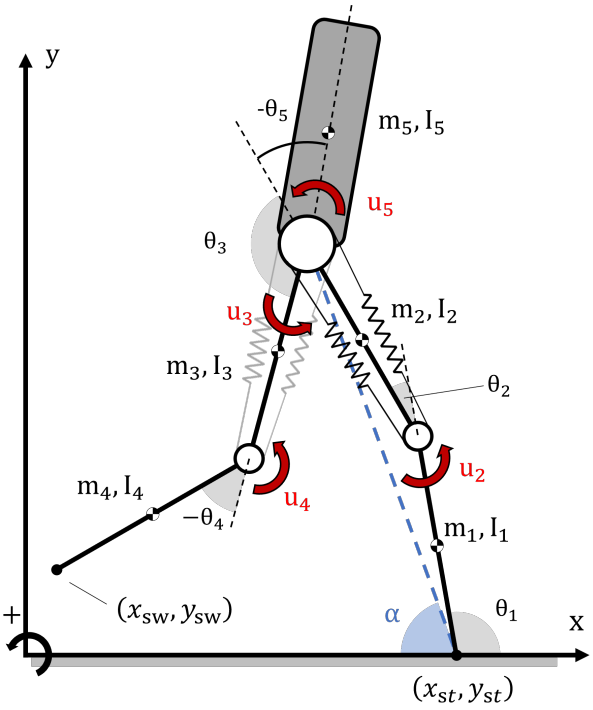


Figure 6.1: 5-link underactuated bipedal robot model where the inputs are indicated with red arrows

terrain blindly, it would ease the burden on the controller of handling this issue and reduce the problems caused by inaccurate perception. In [34], a one step adaptation strategy based on an actuated dual-SLIP template model was proposed, so that humanoid robot can walk on rough terrain without perceiving it for terrain variations of ±5 cm.

One way to tackle the robustness issue is to do it on a mechanical level, namely by adding compliant elements. For example in [35], they found that the series elastic compliance has a stronger impact compared to the parallel one. Also, it is indicated that a stiffer leg increases efficiency but reduces the robustness.

However, compliant elements are usually implemented in monoarticular fashion or as series elastic actuation. Monoarticular muscles are connected to two links and can drive a single joint. On the other hand, biarticular muscles are connected to two links separated by a third one and they can drive two joints at the same time [19]. There are studies that indicate the benefits of these under-studied biarticular muscles.

In [36], the important contributions of biarticular muscles to trajectory control, stiffness control, and output control that take place at the extremities were illustrated. In [37], it is shown that the biarticular muscle mechanism they use in the robotic leg contributes to improved force capacity in such a way that the total output performance is maintained while individual actuator requirements are reduced. It was shown in [21] that biarticular muscles contribute to muscle coordination when performing a jumping motion.

We showed in a previous study that walking efficiency, walking speed and minimum input torque requirements can be significantly improved by using passive biarticular muscles [29]. This study was conducted using trajectory optimization methods and it illustrated that different stiffness values were needed for different optimization goals.

This chapter focuses on terrain-blind walking based on optimized trajectories of a 5-link underactuated (point-footed) biped robot equipped with biarticular muscles. The primary contributions of this chapter includes:

- Proposing a controller that can handle walking on a terrain with random height variations using reference trajectories without perception/sensing and validating it on simulation experiments. We achieved this by using different reference trajectories for walking and stepping-down.
- Investigating the effects of passive biarticular muscles on a bipedal robot's ability to walk in rough terrain and how different physical parameters of these biarticular springs can effect the robustness.

This chapter is organized as follows. Section 6.1 describes the robot model. Section 6.2 describes the optimization setup that was used to obtain the reference trajectories. Section 6.3 describes the proposed controller and in Section 6.4 we present our results.

6.1 Systems and Modeling

In this section, we will describe the 5-link underactuated bipedal robot model used in this chapter and our method of generating random rough terrains.

6.1.1 Bipedal Walker with Biarticular Spring

The robot model and the notations that are used to describe it can be seen in Figure 6.1. This planar model consists of 5 links representing the lower leg, the upper leg and the torso. Model is underactuated and point-footed (no ankle torque). It has 2 actuators on the knees and 2 on the hip where the revolute joints are positioned. Also, there are springs connected in a biarticular configuration between the torso and the lower legs.

Equation of motion of this model can be written as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{S}\mathbf{u} + \boldsymbol{\tau}(\mathbf{q}), \tag{6.1}$$

where $\mathbf{q} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^{\mathrm{T}} \in \mathbb{R}^5$ are the generalized coordinates, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{5\times5}$ is the inertia matrix, $\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^5$ is the Coriolis, centrifugal and gravitational terms vector, $\mathbf{S} \in \mathbb{R}^{5\times4}$ is the distribution matrix of the inputs, $\mathbf{u} = [u_2, u_3, u_4, u_5]^{\mathrm{T}} \in \mathbb{R}^4$ are the input torques and $\boldsymbol{\tau} \in \mathbb{R}^5$ represents the torques generated by the biarticular springs. $\boldsymbol{\tau}$ can be expanded as:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{\mathrm{st}} + \boldsymbol{\tau}_{\mathrm{sw}}, \tag{6.2}$$

where subscripts “st" and “sw" respectively represent the “stance leg" which is the leg that is in contact with the ground and the other leg called the “swing leg".

We calculate the biarticular spring torques in the same manner as [29] where the partial derivative of the potential energy stored in the springs is taken with respect to the generalized coordinates, resulting in:

$$\boldsymbol{\tau}_{\mathrm{st}} = \begin{bmatrix} 0 \\ -\kappa\, r_{\mathrm{k}} \Delta l_{\mathrm{st}} \\ 0 \\ 0 \\ -\kappa\, r_{\mathrm{h}} \Delta l_{\mathrm{st}} \end{bmatrix}, \; \boldsymbol{\tau}_{\mathrm{sw}} = \begin{bmatrix} 0 \\ 0 \\ \kappa\, r_{\mathrm{h}} \Delta l_{\mathrm{sw}} \\ \kappa\, r_{\mathrm{k}} \Delta l_{\mathrm{sw}} \\ -\kappa\, r_{\mathrm{h}} \Delta l_{\mathrm{sw}} \end{bmatrix}, \tag{6.3}$$

where κ [N/m] is the biarticular spring stiffness, r_{h} [m] and r_{k} [m] are the lever arm lengths with subscripts “h" and “k" referring to “hip" and “knee".

$$\Delta l_n = r_h(\varphi_h^n - \varphi_{h0}) - r_k(\varphi_k^n - \varphi_{k0}),\; n \in \{\mathrm{sw}, \mathrm{st}\}, \tag{6.4}$$

are the deflection of the respective spring. Terms related to biarticular springs are indicated in Figure 4.3.

We introduce the lever arm ratio $r = r_h/r_k$, and new spring constant term $\bar{\kappa} = \kappa\, r_{\mathrm{k}}^2$ and deflection:

$$\Delta \bar{l}_n = \frac{\Delta l_n}{r_{\mathrm{k}}}, \tag{6.5}$$

so that in Section 6.2, we can search for r and $\bar{\kappa}$ [Nm] instead of the r_{h}, r_{k} and κ [N/m]. Equation (6.3) can be rewritten using these new terms.

Equation (6.1) models the single stance phase (when one foot is on the ground and the other is doing the swinging motion). When the swing foot contacts the ground (touch-down), an impact occurs and model goes to a double stance phase where both foot are on the ground. In this chapter, we assume an instantaneous double stance phase i.e. swing leg and stance leg switch instantaneously at the moment of the impact and after the impact, swing leg lifts up from the ground without interaction (lift-off). The reset map is given by:

$$\boldsymbol{x}^+ = f_H(\boldsymbol{x}^-). \tag{6.6}$$

where $\boldsymbol{x} = [\mathbf{q}^{\mathrm{T}}, \dot{\mathbf{q}}^{\mathrm{T}}]^{\mathrm{T}}$. During this event, position of the robot remains the same, only the swing leg and stance leg are swapped but velocities change discontinuously [29].

In this chapter, we also compare the performance of our model with the default bipedal model (model without the biarticular springs). Default model can be achieved by setting

the spring stiffness values to zero in Equation 6.1. Throughout this chapter, the surface is considered rigid with sufficient friction to allow the movement.

6.1.2 Walking on soft ground

We also wanted to test our proposed controller on "soft" ground. Many difficult terrains can be not only non-even but also soft. To do this, we need to use "floating base" coordinate system:

$$\mathbf{q}_\mathrm{f} = [x_\mathrm{st}, y_\mathrm{st}, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^\mathrm{T} \in \mathbb{R}^7 \tag{6.7}$$

where x_st and y_st are the horizontal and vertical positions of the stance foot. Equation of motion of the floating base system can be written as:

$$\mathbf{M}_\mathrm{f}(\mathbf{q}_\mathrm{f})\ddot{\mathbf{q}}_\mathrm{f} + \mathbf{H}_\mathrm{f}(\mathbf{q}_\mathrm{f}, \dot{\mathbf{q}}_\mathrm{f}) = \mathbf{S}_\mathrm{f}\mathbf{u} + \boldsymbol{\tau}_\mathrm{f}(\mathbf{q}_\mathrm{f}) + \mathbf{J}^\mathrm{T}\boldsymbol{\lambda}, \tag{6.8}$$

where $\mathbf{J}$ is the Jacobian matrix that maps the generalized coordinates to the stance foot horizontal and vertical position and $\boldsymbol{\lambda}$ are the forces that act on the stance foot. Rest of the terms are the modified versions of those described in Equation (6.1). Given

$$\boldsymbol{\lambda} = [\lambda_x, \lambda_y]^\mathrm{T}, \tag{6.9}$$

we set λ_x to be the constraint force that stops the foot from slipping in the horizontal direction and the reaction force of the "soft" ground acts in the horizontal direction. We modeled the soft ground force as:

$$\lambda_y = \max(\mathrm{k}_\mathrm{ground}(y_\mathrm{ground} - y_\mathrm{st}), 0) - \mathrm{d}_\mathrm{ground}\dot{y}_\mathrm{st} \tag{6.10}$$

where $\mathrm{k}_\mathrm{ground}$ is the stiffness and $\mathrm{d}_\mathrm{ground}$ is the damping of the ground. y_ground is the ground height and it is equal to 0 for the flat terrain and can vary for the rough terrain.
In section 6.1.1, the contact with the ground was modeled as inelastic collision, i.e. the swing foot came to a stop instantaneously after the moment of impact. However, for the soft ground model, impact is modeled as elastic.

6.1.3 Generating the rough terrain

We randomly generated rough terrains to test the performance of the proposed controller. First, numbers between 0 and 1 were randomly generated for every 0.1 [m] interval of the track length to generate a seed. Then the seed is multiplied with $\delta \in [0 : 0.001 : 0.1\,[m]]$ to set the maximum height of the terrain where $\delta = 0$ [m] is the flat terrain. Some sample terrains generated this way can be seen in Fig 6.2. We linearly interpolate for intermediary points in the terrain.

By increasing δ we can monotonically increase the difficulty of a certain terrain seed. This means that the controller will be able to handle the terrain with increasing δ. How the

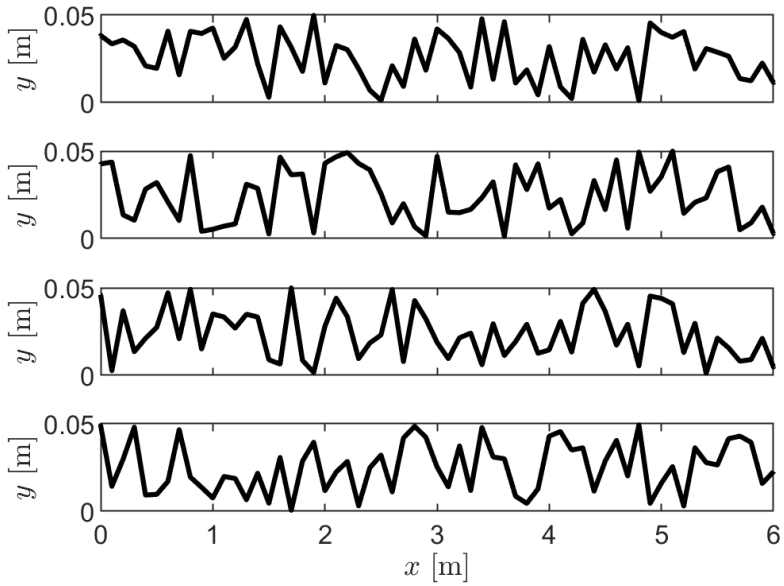


Figure 6.2: Sample of randomly generated rough terrains ($\delta = 0.05$ [m])

terrain looks for different values of δ can be seen in Figure 6.3.

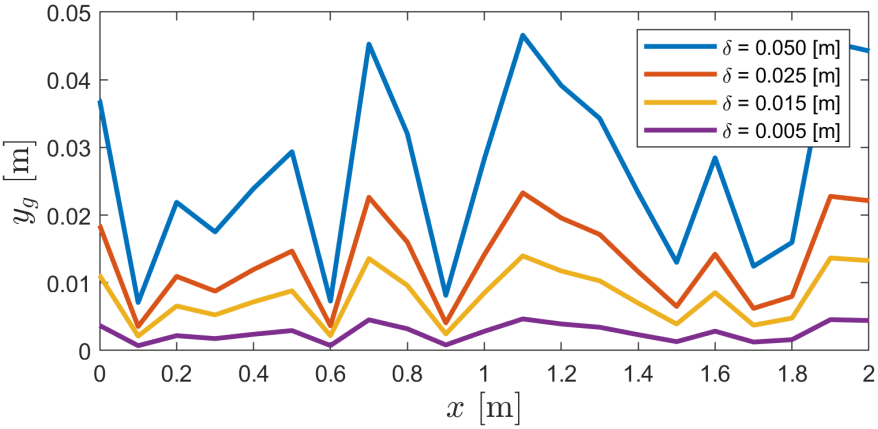


Figure 6.3: Effect of increasing δ on the terrain

6.2 Optimization

In this section, we will describe the optimization setup that was used in obtaining the reference trajectories for regular walking and stepping-down motions using direct collocation method [9]. These methods turn the continuous time problem into discrete one which then can be handled by nonlinear programming solvers. In this chapter, OpenOCL [10] was used to solve the trajectory optimization problem.

Optimization problem can be formulated as

$$\begin{aligned} \min_{\boldsymbol{x},\mathbf{u},\boldsymbol{p},T} \quad & \int_0^T L(\boldsymbol{x}(t),\mathbf{u}(t),\boldsymbol{p})dt \\ \text{s.t.} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t),\mathbf{u}(t),\boldsymbol{p}) \\ & \boldsymbol{r}(\boldsymbol{x},t,\boldsymbol{p}) \leq 0, \end{aligned} \tag{6.11}$$

where $t \in [0,T]$ is the time, $\boldsymbol{x}(t)$ is the state trajectory as defined in Section 6.1.1, $\mathbf{u}(t)$ are the inputs, $\boldsymbol{p}$ are the parameters, $L(\boldsymbol{x}(t),\mathbf{u}(t),\boldsymbol{p})$ is the path cost function, $\boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{p})$ is the system dynamics function (differential equation) and $\boldsymbol{r}(\boldsymbol{x},t,\boldsymbol{p})$ are the constraint functions. The dynamic constraints and the inequality constraints are realized on grid points (collocation points). Number of the grid points was chosen as $N = 24$ and degree of interpolating polynomial as $d = 3$ for the optimization.

We want to obtain two different trajectories for walking on rough terrain. First one is called the “walking trajectory" and this is the reference for our controller most of the time. This would be the only trajectory needed if the gait was to be performed on an even terrain. However, while walking on uneven terrain and when the robot reaches the end of the provided reference walking trajectory without managing to touch the ground due to terrain roughness, we need an additional reference trajectory to safely land the robot. To address this, we will also generate a “stepping-down" trajectory. If the next touch-down position is higher than the ground level, “walking trajectory" can handle it to a certain degree.

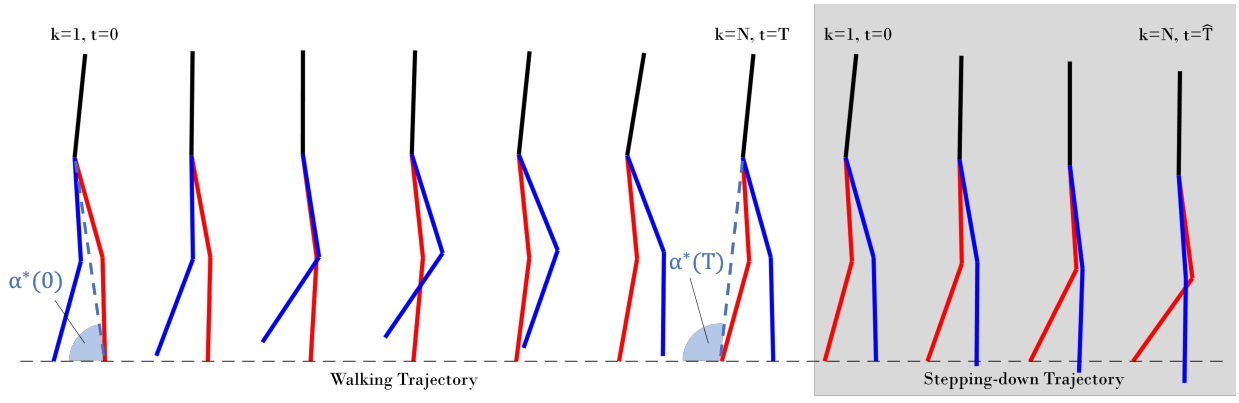


Figure 6.4: Snap-shots of resulting walking trajectory and stepping-down trajectory where red links are the stance leg and blue links are the swing leg

The cost function for the walking and stepping-down trajectories are identical and set to

$$L(\boldsymbol{x}(t), \mathbf{u}(t), \boldsymbol{p}) = \mathbf{u}(t)^{\mathrm{T}}\mathbf{u}(t). \tag{6.12}$$

For the two trajectories, we need to determine a set of constrains so that resulting trajectory is a human-like walking gait. Some constraints for the two trajectories are different but the common ones are:

- Constraining the relative knee joint angles to achieve human-like gaits: $5^\circ < \theta_2 < 22.5^\circ$, $270^\circ < \theta_4 < 345^\circ$
- Upper body must remain straight: $80^\circ < \theta_5 < 90^\circ$
- The angular velocity of motors must not be over the desired limit: $|\dot{\theta}_i| < 10$ [rad/s], $i \in \{1, 2, 3, 4, 5\}$
- Center of mass of the robot should always be moving with positive velocity in the x direction: $\dot{x}_{\mathrm{CoM}}(t) > 0$ [m/s]
- Setting a lower bound for virtual stance leg angle α's velocity to keep it monotonically increasing (we set the lower bound to 0.3 [rad/s] rather than to 0 [rad/s] in order to have some safety margin in the case of blind walking in rough terrain): $\dot{\alpha} > 0.3$ [rad/s]
- The step length of the robot is set to be 0.25 [m]

Snap-shots from resulting walking and stepping-down trajectories are shown in Figure 6.4.

6.2.1 Walking Trajectory

The constraints for the walking trajectory are set as follows:

- Swing foot related constraints: $y_{\mathrm{sw}}(0) = y_{\mathrm{sw}}(T) = 0$ [m], $y_{\mathrm{sw}}(0 < t < T) > 0$ [m], $\dot{y}_{\mathrm{sw}}(T) < -0.2$ [m/s], $\dot{x}_{\mathrm{sw}}(T) < 0$ [m/s], $\dot{x}_{\mathrm{sw}}(t < T) > 0$ [m/s]
- The trajectory must be periodic: $\boldsymbol{x}(0) = f_H(\boldsymbol{x}(T))$ (f_H is the reset map in Equation 6.6)
- Mechanical parameters of the biarticular springs were constrained as: $0.01 \leq r \leq 5$ and 0 [Nm] $\leq \bar{\kappa} \leq 2000$ [Nm].
- Swing foot must avoid a virtual elliptic obstacle (prevents foot dragging and keeps to swing leg from contacting the ground early on rough terrains):

$$\left(\frac{x_{\mathrm{sw}}(t) - d_{\mathrm{obs}}}{w_{\mathrm{obs}}}\right)^2 + \left(\frac{y_{\mathrm{sw}}(t)}{h_{\mathrm{obs}}}\right)^2 \geq 1, \tag{6.13}$$

  where x_{sw} and y_{sw} are horizontal and vertical positions of the swing foot, $d_{\mathrm{obs}} = 0$ [m] is the horizontal position of the elliptic obstacle (from the stance foot), $w_{\mathrm{obs}} = 0.2$ [m] and $h_{\mathrm{obs}} = 0.05$ [m] are the width and height of the ellipse.

Optimization variables for this trajectory are $\boldsymbol{x}$, $\dot{\boldsymbol{x}}$, T, $\boldsymbol{u}$, r and $\bar{\kappa}$.

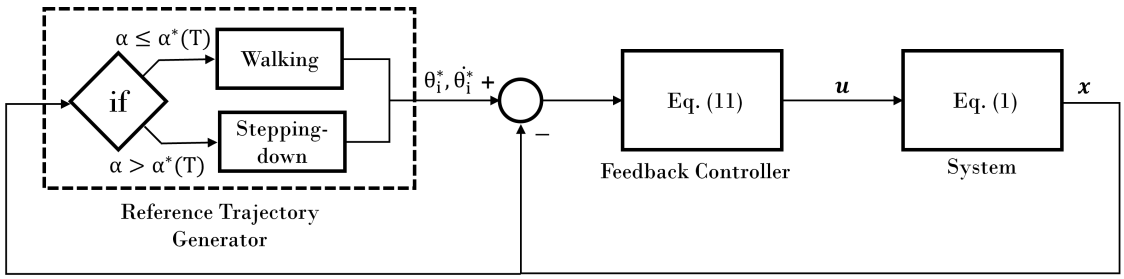


Figure 6.5: The controller diagram

6.2.2 Stepping-down Trajectory

The stepping-down trajectory takes over where the walking trajectory left off at $t = T$ and does a stepping down motion while trying to keep the step length the same. The constraints for the stepping-down trajectory are set as follows (terms related to this trajectory are indicated with $\hat{\text{L}}$ notation):

- Stepping-down trajectory continues from the end point of the walking trajectory: $\hat{\boldsymbol{x}}(0) = \boldsymbol{x}(T)$
- Swing foot related constraints: -0.11 [m]$\leq \hat{y}_{\text{sw}}(\hat{T}) \leq -0.1$ [m], $\dot{\hat{y}}_{\text{sw}}(\hat{T}) < -0.2$ [m/s], $\dot{\hat{x}}_{\text{sw}}(\hat{T}) < 0$ [m/s]

This trajectory uses the same biarticular spring parameters obtained in "walking trajectory" optimization and variables are $\hat{\boldsymbol{x}}$, $\dot{\hat{\boldsymbol{x}}}$, $\hat{T}$, $\hat{\boldsymbol{u}}$. Here we set the step down height to be about 10 cm which can be adjusted depending on the task or terrain.

6.3 Control

In this section, we will introduce a controller that can handle blind walking on uneven terrain using the reference trajectories generated by direct collocation optimization. We will use a feedback linearization scheme for trajectory tracking. The model has 5 degrees of freedom but only 4 joints are controlled (point-foot model) resulting in underactuation. Because of this, we will formulate all the reference trajectories as a function of the virtual stance leg angle α as the phase variable (Figure 6.1) to stabilize the system. Phase variable based implementations have been shown to be more robust compared to time based ones [31].

In the optimization part, α was constrained to be monotonically increasing. Reference trajectories are indicated by the * term where $\theta_i^*(\alpha)$ is the reference joint angle and $\dot{\theta}_i^*(\alpha)$ is the reference angular velocity for the i^{th} joint. Since direct collocation just outputs the results at collocation points, we linear interpolate for the in-between values.

The single stance phase ends when the swing foot contacts the ground (touch-down) and begins when it ceases contact with it (lift-off). In this chapter, double stance phase is an

instantaneous event as mentioned in Section 6.1.

Reference walking trajectory obtained in Section 6.2 is defined in $\alpha \in [\alpha^*(0), \alpha^*(T)]$ where $\alpha^*(T)$ is the virtual stance leg angle at the end of the trajectory where swing foot contacts the ground. Using a predefined trajectory for walking on rough terrain is tricky because reference is defined for a certain range of α. If the α is within the defined range in the beginning of the single stance phase, reference walking trajectory can be used without any modifications. However, the single stance phase could begin with a lift-off virtual stance leg angle that is out of the defined range ($\alpha^{\mathrm{LO}} < \alpha^*(0)$ or $\alpha^{\mathrm{LO}} > \alpha^*(T)$) because of the random ground height. For the $\alpha^{\mathrm{LO}} < \alpha^*(0)$ case, we use the modified α,

$$\begin{aligned} \hat{\alpha} &= \alpha + a\alpha + b \\ a &= \frac{\alpha^*(0) - \alpha^{\mathrm{LO}}}{\alpha^{\mathrm{LO}} - \alpha^{\mathrm{merge}}}, \\ b &= -a\alpha^{\mathrm{merge}} \end{aligned} \tag{6.14}$$

for obtaining reference trajectories $\theta_i^*(\hat{\alpha})$, $\dot{\theta}_i^*(\hat{\alpha})$. This makes sure α is in the defined range and linearly merges to the original one at α^{merge}. α^{merge} was chosen as the middle collocation point ($k = 12$). If $\alpha^{\mathrm{LO}} > \alpha^*(T)$ at the beginning of the single stance phase, the gait is considered to have failed.

Another difficulty that can occur when walking on rough terrain is when the robot reaches the end of the reference walking trajectory but the expected touch-down condition ($y_{\mathrm{sw}} \leq y_{\mathrm{ground}}$) is not satisfied ($\alpha > \alpha^*(T)$). This is when the reference trajectory switches from "walking" to "stepping-down" and just tries to reach the ground with the swing foot while keeping the same step-length (and satisfying the other constraints).

A diagram of the controller can be seen in Figure 6.5. The reference trajectory generator sends the appropriate robot configuration as the reference according to the current α. We use feedback linearization to track actuated joint trajectories. Inputs can be chosen as:

$$\mathbf{u} = (\mathbf{T}\mathbf{M}^{-1}\mathbf{S})^{-1}(\boldsymbol{v} + \mathbf{T}\mathbf{M}^{-1}(\mathbf{H} - \boldsymbol{\tau})), \tag{6.15}$$

to linearize the system given in Equation (6.1) where $\mathbf{T} \in \mathbb{R}^{4\times 5}$ is the task space matrix that maps the generalized coordinates to the actuated ones and

$$\boldsymbol{v} = \mathrm{K_p}\boldsymbol{y} + \mathrm{K_d}\dot{\boldsymbol{y}}, \tag{6.16}$$

$$\boldsymbol{y} = \begin{bmatrix} \theta_2^*(\alpha) - \theta_2 \\ \theta_3^*(\alpha) - \theta_3 \\ \theta_4^*(\alpha) - \theta_4 \\ \theta_5^*(\alpha) - \theta_5 \end{bmatrix}. \tag{6.17}$$

$\mathrm{K_p}$ and $\mathrm{K_d}$ are the proportional and derivative gains and are set to same values for each actuated joint.

Table 6.1: 5 Link Model Parameters

$l_1 = l_4 : 0.48$ [m]	$l_2 = l_3 : 0.48$ [m]	$l_5 : 0.48$ [m]
$m_1 = m_4 : 5$ [kg]	$m_2 = m_3 : 5$ [kg]	$m_5 : 60$ [kg]
$I_i = m_i l_i^2/12$ [kg $\cdot$ m^2], $i = 1, 2, 3, 4, 5$		

6.4 Results and Discussion

In this section, we will present the results when the trajectories obtained in Section 6.2 are used in the controller proposed in Section 6.3 on the robot model described in Section 6.1, where biarticular muscle parameters are set to those obtained by direct collocation optimization. We also investigate the effects of different biarticular muscle parameter combination's effect on robustness.

Simulations were performed in Matlab SIMULINK environment with variable step ode45 solver, max step size of 1e-3 and an absolute tolerance of 1e-8. The physical parameters of the robot are provided in Table 6.1.

6.4.1 Walking on flat terrain

Figure 6.6 shows the proposed controller's performance on a flat terrain. We can see that tracking performance is quite good for this under-actuated system. Gains were set to $\mathrm{K_p} = 9700$ and $\mathrm{K_d} = 220$ for this simulation. As mentioned in Section 6.2, biarticular spring parameters were also optimized as they were also set as optimization parameters. We set them to resulting $r = 1.8173$ and $\bar{\kappa} = 29.8220$ [Nm] values for this gait. Average velocity was 0.61 [m/s].

6.4.2 Walking on rough terrain

Now, we will present the performance of the controller when the 5-link model is set to walk blindly (without any information of the terrain height changes) on the randomly generated terrains described in Section 6.1.3. Gains and the biarticular spring parameters are kept the same as those mentioned in Section 6.4.1. Terrain difficulty was set to $\delta = 0.05$ [m].

Figure 6.7 shows some snap shots from this gait. Trajectory tracking performance of the proposed controller can be seen in Figure 6.8. Trajectory tracking performance is still quite good considering the unknown terrain height changes the robot has to handle. It can be seen that the duration of a single step is not constant anymore and depends on the terrain. Figure 6.9 shows the trajectory of the virtual stance leg angle during this gait. In this figure, it is easier to see that the duration of a step is not constant. Also, we can see that α can go out of the reference trajectory bounds (shown by the dotted lines). When α goes above the dotted line ($\alpha > \alpha^*(T)$), the controller switches the reference to stepping-down trajectory and when α starts below the dotted line, the modification described in Equation 6.14 is

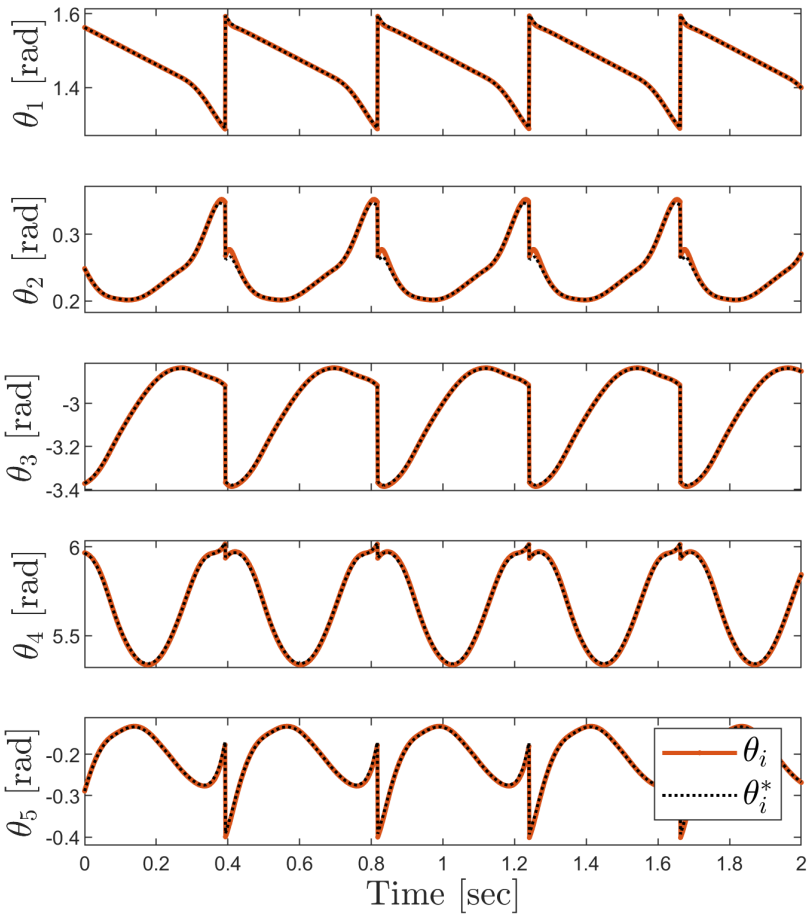


Figure 6.6: Trajectory tracking results on flat terrain

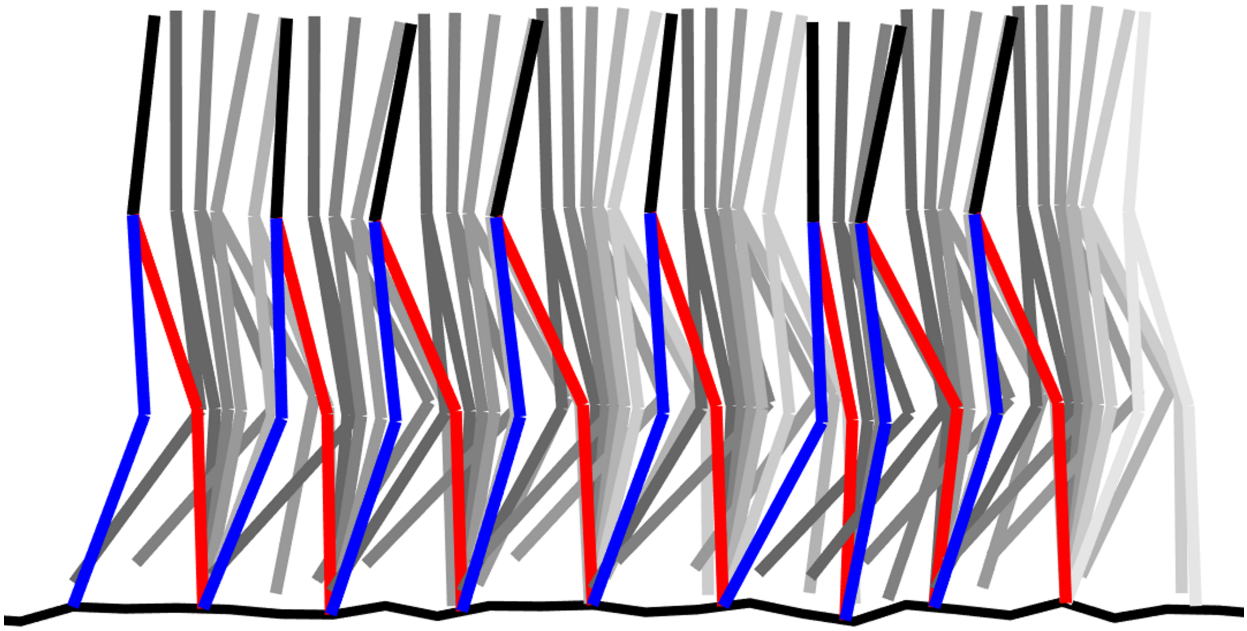


Figure 6.7: Snap-shots from walking on a rough terrain with $\delta = 0.05$ [m] (Biarticular muscles were not shown on this figure to reduce visual clutter)

triggered. Overall, we can see that the proposed controller can handle walking on random rough terrain with good trajectory tracking performance. Average velocity was 0.442 [m/s] for this gait.

6.4.3 Effect of biarticular muscle parameters on rough terrain walking

We wanted to see the effect of different biarticular spring parameters on robustness, especially for the blind-walking on rough terrain case. In our model, biarticular springs have two parameters that can be adjusted: the lever arm ratio r and spring constant $\bar{\kappa}$.

Our metric of robustness is the maximum δ parameter the robot can handle, i.e. $\bar{\delta}$. We introduced how the random rough terrain was generated in Section 6.1.3 and terrain difficulty increased as δ parameter was increased. Figure 6.10 shows the resulting $\bar{\delta}$ values in a 3D plot for combinations of $r = [1, 1.2, 1.4, ..., 5]$ and $\bar{\kappa} = [0, 5, 10, 15, ..., 200]$ [Nm]. For each combination, we start with $\delta = 0$ (flat terrain) and run the walking simulation. If the robot is successful at walking in the terrain for 10 seconds without falling, it is considered a successful walk and we increase δ by 0.001 [m] and run the simulation again. We do this until the robot can't handle the terrain difficulty anymore and the maximum δ value it can handle is recorded as $\bar{\delta}$ and is shown in the figure. Figure 6.10 is the average result of simulation experiments in 10 different random terrain (4 of them are shown in Figure 6.2).

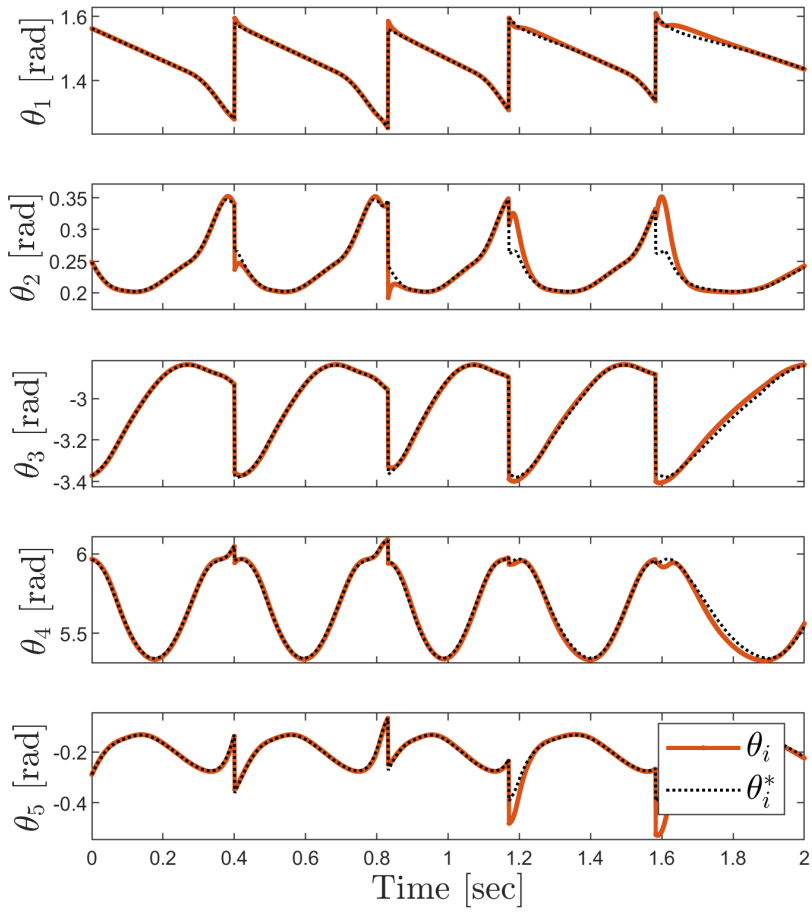


Figure 6.8: Trajectory tracking results on rough terrain with $\delta = 0.05$ [m]

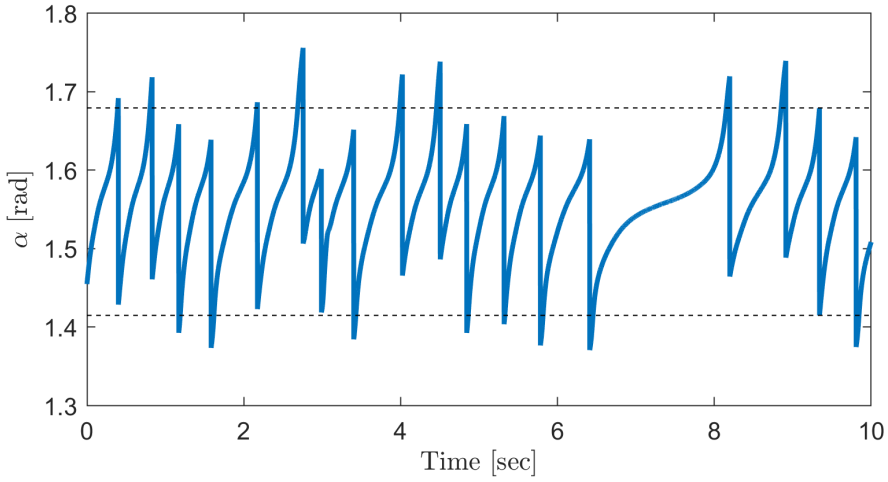


Figure 6.9: Virtual stance leg angle α when walking on rough terrain with $\delta = 0.05$ [m]. The dotted lines show the upper and lower limits of the reference walking trajectory $\alpha^* \in [\alpha^*(0), \alpha^*(T)]$

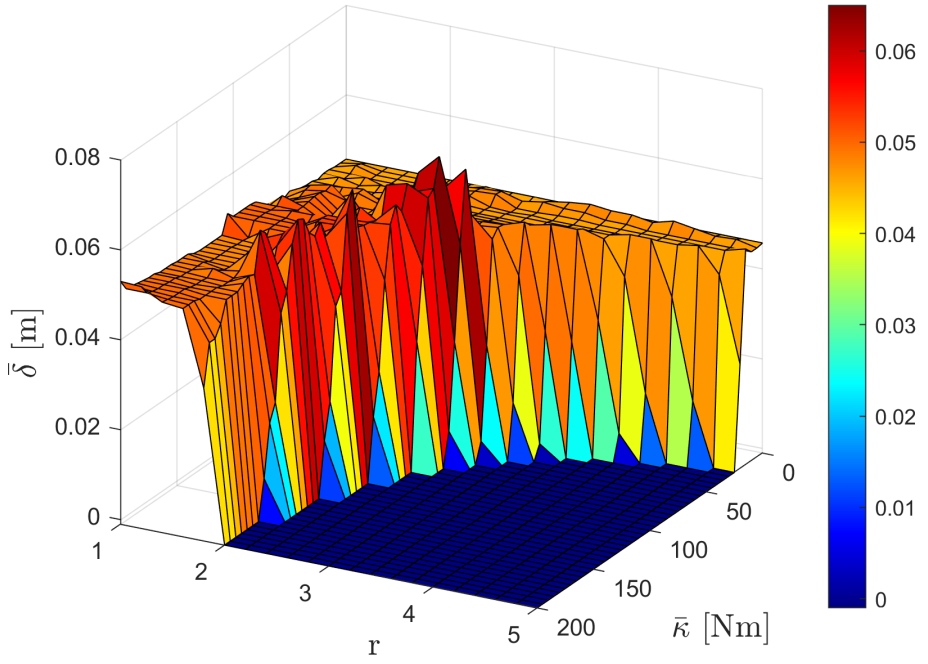


Figure 6.10: Effects of different biarticular spring parameter combinations on robustness to terrain difficulty. Vertical axis shows $\bar{\delta}$ which is the maximum terrain difficulty the model with indicated parameter combination can handle. Larger $\bar{\delta}$ means it is more robust.

In Figure 6.10, $\bar{\kappa} = 0$ corresponds to a model without biarticular springs (we call this the default model) where $\boldsymbol{\tau} = 0$ in Equation (6.1). We can see that by adding biarticular springs, the robot can handle rougher terrains for some spring parameter combinations. The maximum terrain roughness that the default model could handle was $\delta = 4.49$ [cm]. It can be seen that there are a lot of r and $\bar{\kappa}$ combinations that can surpass this value and make the system more robust for blind-walking on rough terrain. The maximum value was reached with a parameter combination of $r = 2.2$, $\bar{\kappa} = 105$ [Nm], the robot was able to handle a terrain with $\delta = 6.47$ [cm] which is a 44.098% increase compared to the default model. Figure 6.11 shows the same results for fixed r values in a 2D plot which is a bit more easier to read.

These are the average values for walking on different random terrains. For one of the terrains, robot was able to increase its $\bar{\delta}$ value from 0.0500 [m] for the default model to 0.0960 [m] by using biarticular muscles with $r = 2$, $\bar{\kappa} = 130$ [Nm] which is a 92% increase. The smallest maximum increase for a single terrain was from $\bar{\delta} = 0.05$ [m] for the default model to $\bar{\delta} = 8.3$ [cm] for the model with biarticular muscles which is a 66% increase. We can see that adding biarticular springs always ended up increasing the robustness for different random terrain. The reason for values for single terrains being larger than the average value is that different biarticular spring parameter settings perform better for different terrain.

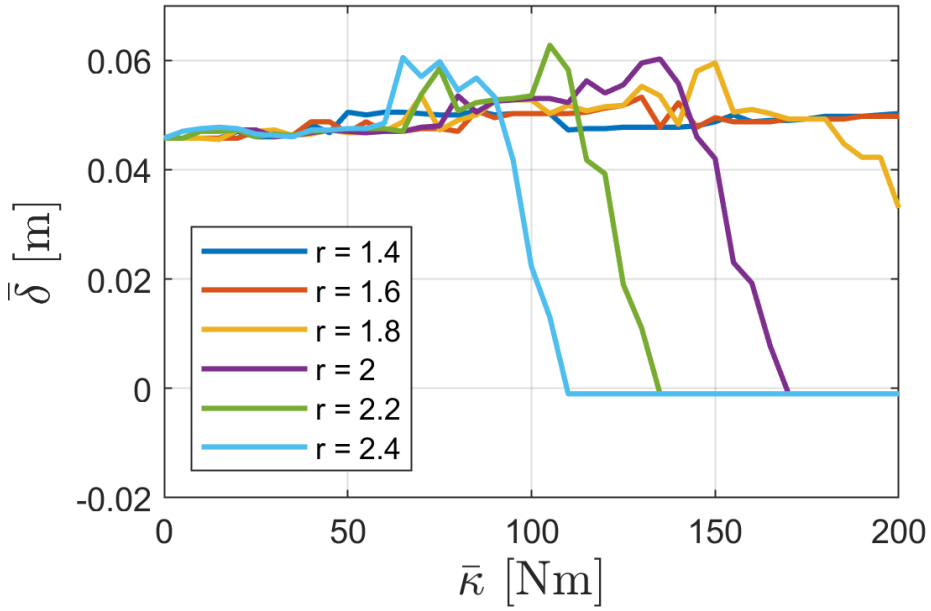


Figure 6.11: This figure shows the same results presented in Figure 6.10 but for fixed r values

However, we can also see that for larger r and $\bar{\kappa}$ values, the robot can't even walk on even terrain (shown by the dark blue region in Figure 6.10). This is because the biarticular muscles generate large torques that actuators can't handle(there is a $\pm$ 200 [Nm] torque limit on the actuators).

We have also investigated the efficiency of walking for different biarticular spring parameters that was used in this chapter. Specific resistance (SR) [38]

$$SR := \frac{p}{Mgv},\ p = \frac{1}{T}\int_0^T \sum_{i=2}^{5} |u_i \omega_i| dx, \tag{6.18}$$

was chosen as the efficiency indicator where T [secs] is the end time of one step, M [kg] is the total weight of the robot, g $[m/s^2]$ is the gravitational term, v $[m/s]$ is the average speed and p $[J/s]$ is the average input energy. A smaller SR value means that the gait is more energy-efficient. When walking on flat ground, $SR = 0.27$ for the default model ($\bar{\kappa} = 0$), $SR = 0.18$ for the BA model with spring parameters obtained from the optimization ($\bar{\kappa} = 29.82$ [Nm], $r = 1.82$) and $SR = 0.65$ for the BA model with most robust spring parameters ($\bar{\kappa} = 105$ [Nm], $r = 2.2$) values were obtained. This shows that for different criterion, best pair of parameters are different. Biarticular muscles can increase the efficiency as shown in [29] but if we want to make the robot more robust, some efficiency must be sacrificed. Best way would be to adjust the parameters according to the environment and the task. For example, $\bar{\kappa}$ can be changed by introducing a stiffness adjustment mechanism as in [39] and r can be adjusted by a variable radius mechanism as in [40].

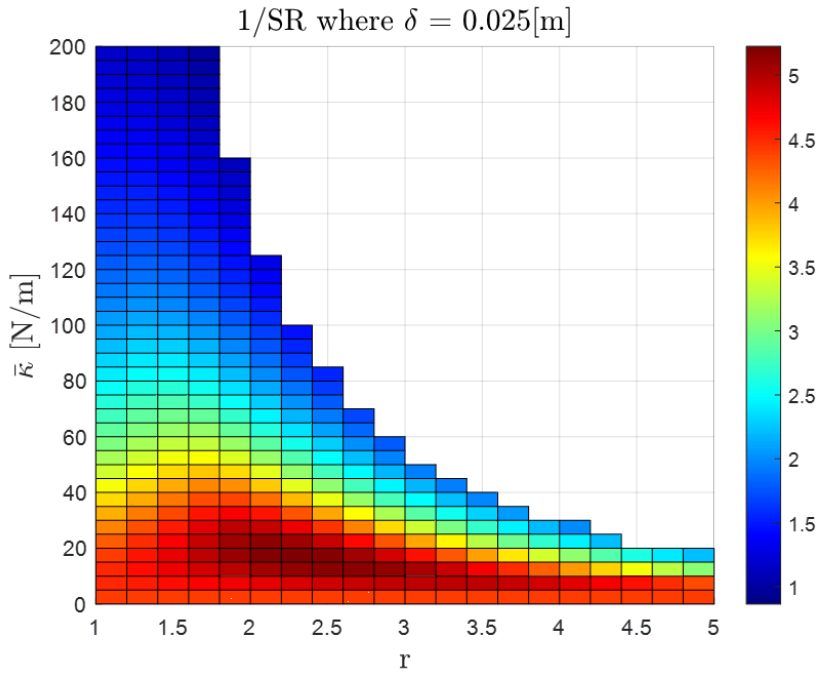


Figure 6.12: 1/SR values for different biarticular spring parameter combinations

6.4.4 Increasing terrain-blind walking and efficiency simultaneously

In Section 6.4.3, we only looked for the biarticular spring parameter combinations that increased the robustness. Figure 6.12 shows 1/SR values for $\delta = 2.5$ [cm]. If we compare this with Figure 6.10, it can be seen that the "robust" regions doesn't necessarily coincide with the "efficient" region.

We can come up with a different criteria to increase robustness and efficiency simultaneously. First, let's define the normalized performance indicators as:

$$\bar{\delta}^* = \frac{\bar{\delta}}{\bar{\delta}(\bar{\kappa} = 0)} \tag{6.19}$$

$$\mathrm{SR}^* = \frac{\mathrm{SR}}{\mathrm{SR}(\bar{\kappa} = 0))}. \tag{6.20}$$

Then, we can come up a criteria that makes sure both the efficiency and robustness is increased the simultaneously compared to default model as:

$$\min(\bar{\delta}^* - 1, \frac{1}{\mathrm{SR}^*} - 1). \tag{6.21}$$

Figure 6.13 shows the combined criteria results. Max value of $\bar{\delta}^* = 1.42$ was reached at $r = 2.2$ and $\bar{\kappa} = 105$ [N/m] as per Section 6.4.3. Max value of $1/\mathrm{SR}^* = 1.18$ is reached where $r = 2.2$ and $\bar{\kappa} = 15$ [N/m]. Maximum value of the combined criteria is 49.18 e-3

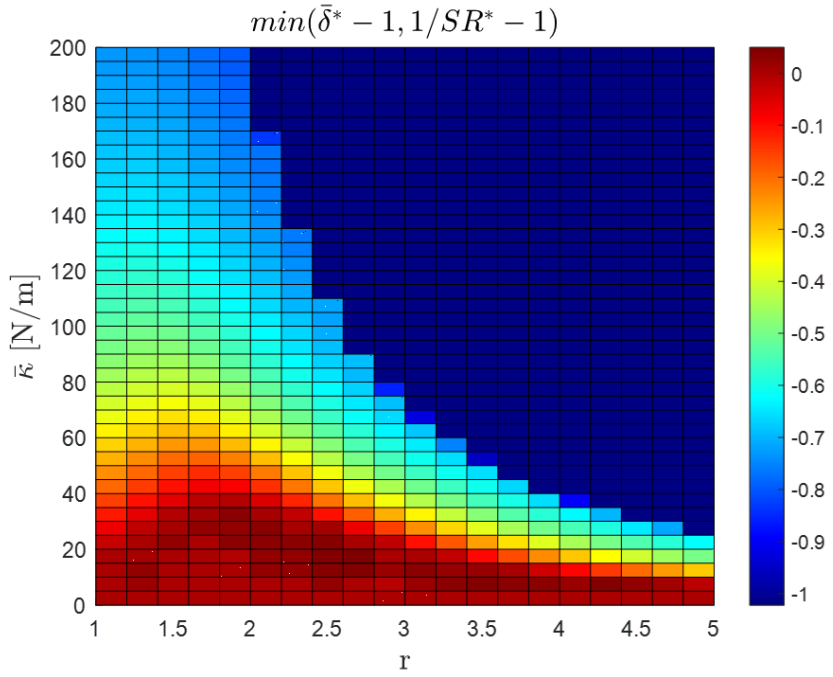


Figure 6.13: Results of the combined criteria given in Equation (6.21)

where $r = 2.6$ and $\bar{\kappa} = 15$ [N/m]. $\bar{\delta}^* = 1.05$ and $\mathrm{SR}^* = 1.15$ at this point.

6.4.5 Walking on soft ground

We designed our controller for rigid uneven terrain but in real life, ground being soft can cause a lot of problems too. We wanted to see if the proposed controller can handle the soft ground case. Figure 6.14 shows the trajectory tracking results and Figure 6.15 shows the stance foot trajectory when walking on soft but flat ground with $\mathrm{k_{ground}} = 90$ [N/mm] and $\mathrm{d_{ground}} = 10$ [Ns/m] parameters. It can be seen that the proposed controller can handle soft ground up to some degree. However, it can't handle softer ground, for example when we set $\mathrm{k_{ground}} = 60$ [N/mm], we couldn't achieve a stable gait. Additional modifications to the controller might be necessary to increase the robustness for the soft ground case.

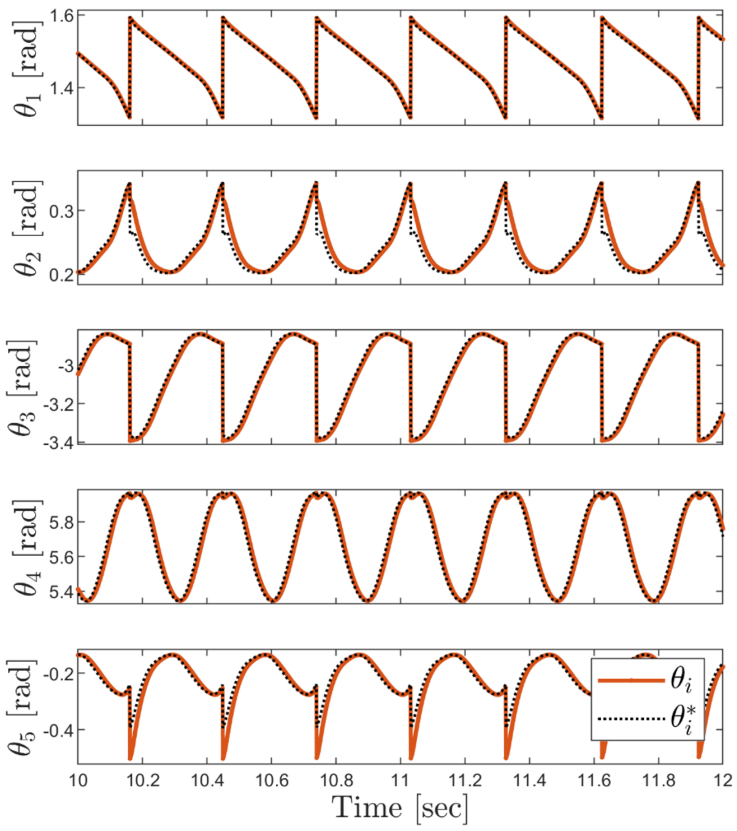


Figure 6.14: Trajectory tracking results for soft ground walking where $k_{ground} = 90$ [N/mm] and $d_{ground} = 10$ [Ns/m]

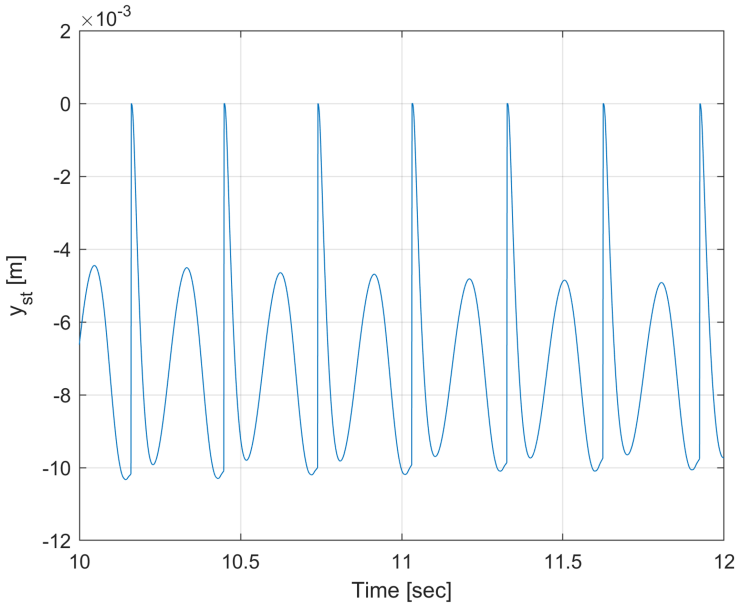


Figure 6.15: Trajectory tracking results for soft ground walking where $k_{ground} = 90$ [N/mm] and $d_{ground} = 10$ [Ns/m]

6.5 Conclusions

In this chapter, we proposed a controller that can use optimized trajectories to traverse through rough terrain without perception/sensing. To achieve this, we used a reference stepping-down trajectory in addition to the reference walking trajectory and also by modifying the phase variable (α) of the reference trajectories when necessary. We also showed how we have obtained these trajectories using direct collocation trajectory optimization. Through simulation experiments, it was shown that a 5-link underactuated biped robot model was able to handle random rough terrain with height changes up to 4.49 cm on average.

Using this controller, we also investigated the effects of passive biarticular muscles on robustness. It was shown that adding biarticular springs can significantly increase the performance for terrain-blind walking on rough terrain. Model with biarticular muscles was able to handle a terrain with 6.47 cm height change on average which is a significant increase compared to the default model. We also investigated how different biarticular spring parameters effect the robustness and found that adding biarticular muscles increased the robustness unless a really stiff spring was chosen or lever arm ratio was set too high. Our study shows that a spring constant of $\bar{\kappa} = 105$ [Nm] and lever arm ratio of $r = 2.2$ gave the best performance.

This study showed that biarticular springs in combination with our proposed controller can handle terrain-blind walking. Being able to handle terrain variations without perception/sensing would truly ease the burden on the high level controller, computation times would decrease and failures due to errors in perception could be mitigated.

As a follow-up to this work, we would like to achieve a velocity tracking scheme with our controller and work on improving the robustness. In [41], we showed that having variable stiffness on a SLIP model can significantly increase the robustness against external pushes. Now with this chapter, we showed that different biarticular muscle parameters may fare better for different terrain and parameters that provide better robustness do not necessarily provide the best walking efficiency. Also in another future study, we would like to see if we can further improve the overall robustness and efficiency by having variable stiffness biarticular muscles and adjustable lever-arm ratio via an adjustment mechanism and an accompanying controller.

Chapter 7

Walking Control of a 5-Link Underactuated Bipedal Robot with Variable Stiffness Biarticular Springs Based VSLIP-SL

Including compliance is crucial for virtually every legged system with multiple moving parts. It plays a vital role in ensuring reliability, inherent stability, safety, efficient energy utilization, and adaptability to various environments. By incorporating compliance, legged systems can enhance their overall performance and address key factors related to their dependability and functionality [42].

Compliance is ubiquitous is biological lifeforms and most of them control the compliance of their muscles to react to certain situations. Humans coactivate theis antagonistic muscles to regulate their impedance [43]. Another study showed that humans learn to stabilize unstable dynamics using the skilful and energy-efficient strategy of selective control of impedance geometry [44].

But is achieving a variable stiffness spring possible? One simple way this can be achieved is by using the variable lever arm principle illustrated in Figure 7.1. The stiffness is adjusted by adjusting the mechanical advantage of a lever arm. Researchers have developed a new actuator with adjustable stiffness called AwAS [45]. This actuator can regulate the joint stiffness through a wide range with minimum energy consumption by means of a small motor.

Another means of achieving variable stiffness is through the use of antagonistically arranged pneumatic artificial muscles. In [46] they were able to control the position and stifness of the ankle joint of an exoskeleton via this sort of actuators.

Biarticular springs were shown to improve a different key walking performance indicators such as walking efficiency, walking speed, minimum torque requirements [29] and walking on rough terrain [47]. Thus, we will try to combine biarticular springs with variable stiffness control. We will focus on developing a controller that can respond to disturbances by

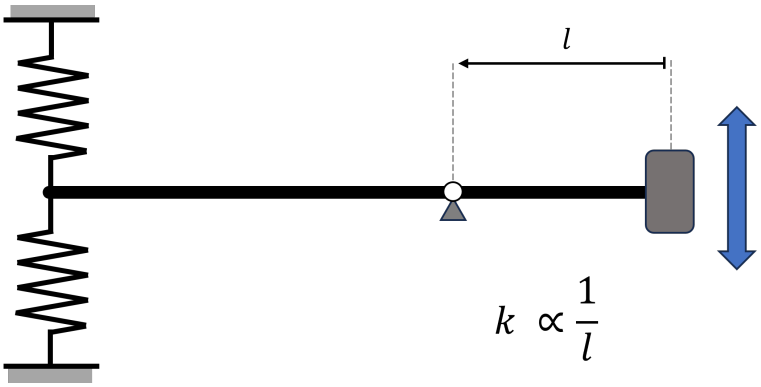


Figure 7.1: Analogy of swinging arms and wobbling mass

changing the stiffness of the legs and not focus on the design of compliant element whose stiffness can be controller. We assume that we can control the stiffness value of the springs without delay and without changing its free position.

In this chapter, we will devise a controller that controls the stiffnesses of biarticular springs to make the biped walker robust against disturbances such as external pushes. This controler will be based on the variable stiffness spring loaded inverted pendulum model with swing leg dynamics (VSLIP-SL) [41]. This template model reacts to the deviations from the originial trajectory by adjust its spring stiffness and we will convert these responses to stiffness change in biarticular springs via a potential energy ratio scheme (Figure 7.2).

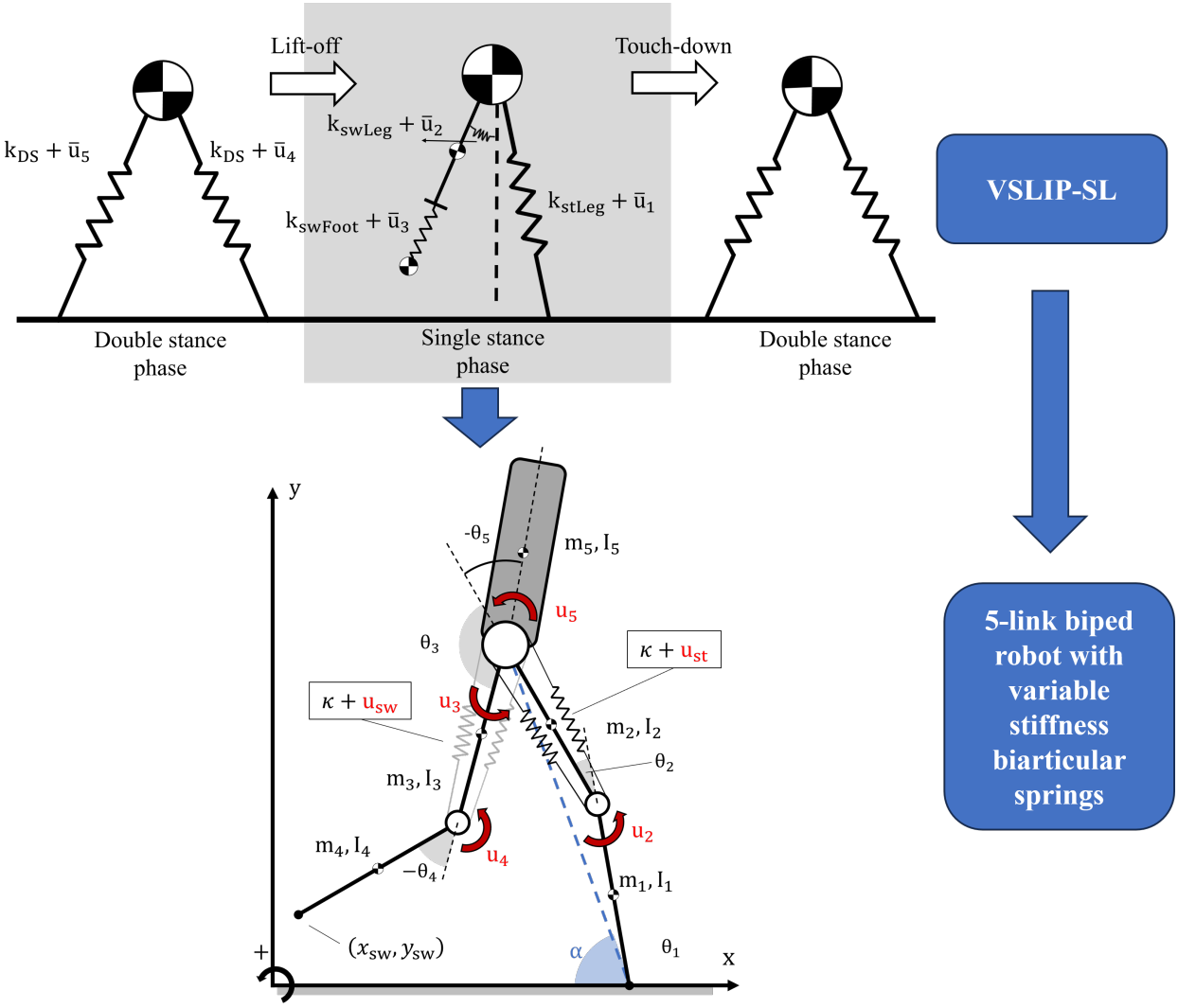


Figure 7.2: Analogy of swinging arms and wobbling mass

7.1 Systems and Modeling

The robot model and the notations that are used to describe it can be seen in Figure 7.2. This planar model consists of 5 links representing the lower leg, the upper leg and the torso. The model has 2 actuators on the knees and 2 on the hip where the revolute joints are positioned. There are springs connected in a biarticular configuration between the torso and the lower legs. We assume that we can change the stiffnesses of these biarticular springs, meaning we have 2 additional inputs, making this system over-actuated.

7.1.1 Single Stance Phase

Equation of motion of this model can be written as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{S}\mathbf{u} + \boldsymbol{\tau}\left(\mathbf{u}_{\mathrm{ba}}\right), \tag{7.1}$$

where,

$$\mathbf{q} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^{\mathrm{T}} \in \mathbb{R}^5, \tag{7.2}$$

are the generalized coordinates, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{5\times5}$ is the inertia matrix, $\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^5$ is the Coriolis, centrifugal and gravitational terms vector, $\mathbf{S} \in \mathbb{R}^{5\times4}$ is the distribution matrix of the inputs,

$$\mathbf{u} = [u_2, u_3, u_4, u_5]^{\mathrm{T}} \in \mathbb{R}^4, \tag{7.3}$$

are the input torques [Nm] and $\boldsymbol{\tau}\left(\mathbf{u}_{\mathrm{ba}}\right) \in \mathbb{R}^5$ represents the torques [Nm] generated by the biarticular springs where

$$\mathbf{u}_{\mathrm{ba}} = \begin{bmatrix} \mathrm{u_{st}} \\ \mathrm{u_{sw}} \end{bmatrix}, \tag{7.4}$$

are the variable stiffness inputs [N/m]. $\boldsymbol{\tau}$ can be expanded as:

$$\boldsymbol{\tau}\left(\mathbf{u}_{\mathrm{ba}}\right) = \boldsymbol{\tau}_{\mathrm{st}}(\mathrm{u_{st}}) + \boldsymbol{\tau}_{\mathrm{sw}}(\mathrm{u_{sw}}), \tag{7.5}$$

where subscripts “st" and “sw" respectively represent the “stance leg" which is the leg that is in contact with the ground and the other leg called the “swing leg".

We calculate the biarticular spring torques in the same manner as [29] where the partial derivative of the potential energy stored in the springs is taken with respect to the generalized coordinates, resulting in:

$$\boldsymbol{\tau}_{\mathrm{st}} = \begin{bmatrix} 0 \\ -(\kappa + \mathrm{u_{st}})\, r_{\mathrm{k}} \Delta l_{\mathrm{st}} \\ 0 \\ 0 \\ -(\kappa + \mathrm{u_{st}})\, r_{\mathrm{h}} \Delta l_{\mathrm{st}} \end{bmatrix}, \ \boldsymbol{\tau}_{\mathrm{sw}} = \begin{bmatrix} 0 \\ 0 \\ (\kappa + \mathrm{u_{sw}})\, r_{\mathrm{h}} \Delta l_{\mathrm{sw}} \\ (\kappa + \mathrm{u_{sw}})\, r_{\mathrm{k}} \Delta l_{\mathrm{sw}} \\ -(\kappa + \mathrm{u_{sw}})\, r_{\mathrm{h}} \Delta l_{\mathrm{sw}} \end{bmatrix}, \tag{7.6}$$

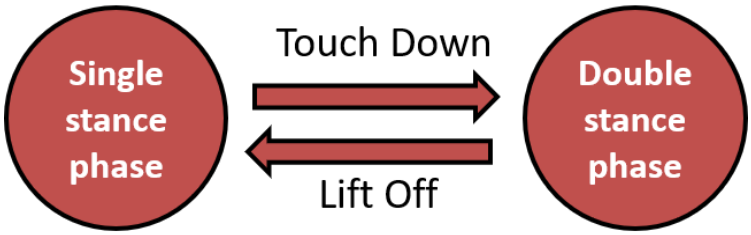


Figure 7.3: Phase transition graph for the variable stiffness BA model

where κ [N/m] is the biarticular spring stiffness, r_{h} [m] and r_{k} [m] are the lever arm lengths with subscripts “h" and “k" referring to “hip" and “knee".

$$\Delta l_n = r_h(\varphi_h^n - \varphi_{h0}) - r_k(\varphi_k^n - \varphi_{k0}),\ n \in \{\mathrm{sw}, \mathrm{st}\}, \tag{7.7}$$

are the deflection of the respective spring. Biarticular springs angles can be calculated as:

$$\begin{aligned} \varphi_h^{\mathrm{sw}} &= \theta_5 - \theta_3 \\ \varphi_k^{\mathrm{sw}} &= \theta_4 \\ \varphi_h^{\mathrm{st}} &= \theta_5 - \pi \\ \varphi_k^{\mathrm{st}} &= \pi - \theta_2 \end{aligned} . \tag{7.8}$$

When the swing foot touches the ground in the single stance phase (the touch down event), the model goes into the double support phase and when the swing foot lifts off from the ground in the double support phase (the lift off event), the model goes back in to the single support phase (Figure 7.3). When the swing leg hits/contacts the ground in the single stance phase, a collision occurs where the generalized momentum of the system changes discontinuously. One way to model this is to assume that an impulse force acts on the system to change velocities discontinuosly while position of the system is not affected. This can be expressed as:

$$\mathbf{M}(\mathbf{q})\Delta\dot{\mathbf{q}} = \mathbf{J}_c^{\mathrm{T}}\lambda_{\mathrm{impact}}, \tag{7.9}$$

where $\mathbf{J}_c \in \mathbb{R}^{2\times 5}$ is a constraint Jacobian matrix that maps the joint velocities to swing foot velocity in horizontal and vertical directions where generalized reaction forces in those directions are given as $\boldsymbol{\lambda}_{\mathrm{impact}} \in \mathbb{R}^2$. Assuming that the impact is inelastic, velocity of the swing foot touching the ground will become zero after the impact which can be expressed as:

$$\mathbf{J}_c(\mathbf{q})\dot{\mathbf{q}}^+ = 0 \Leftrightarrow \mathbf{J}_c(\mathbf{q})\Delta\dot{\mathbf{q}} = -\mathbf{J}_c(\mathbf{q})\dot{\mathbf{q}}^-, \tag{7.10}$$

where superscripts $^-$ expresses the moment just before the impact and $^+$ just after the impact. By solving equations (7.9) and (7.10) together to solve for λ_{impact} we can get:

$$\lambda_{\mathrm{impact}} = -(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^T)^{-1}\mathbf{J}_c\dot{\mathbf{q}}^-. \tag{7.11}$$

By inserting $\lambda_{\mathrm{impact}}$ into equation (7.9), we can obtain:

$$\dot{\mathbf{q}}^+ = (\mathbf{I} - \mathbf{M}^{-1}\mathbf{J}_c^T(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^T)^{-1}\mathbf{J}_c)\dot{\mathbf{q}}^-, \tag{7.12}$$

which gives the joint velocities after the impact. Please note that after the impact, definitions of the legs also change i.e. the swing leg becomes the stance leg and vice versa and the joint angles and velocities are remapped accordingly. The reset map is given by:

$$\boldsymbol{x}^{+} = f_H(\boldsymbol{x}^{-}). \tag{7.13}$$

where $\boldsymbol{x} = [\mathbf{q}^{\mathrm{T}}, \dot{\mathbf{q}}^{\mathrm{T}}]^{\mathrm{T}}$.

7.1.2 Double Stance Phase

After the touch down event, the swing leg becomes the stance leg and the previous stance leg becomes the new swing leg. It is assumed that the new swing foot remains on the ground without sliding until its vertical velocity becomes positive. This non-slip constraint is achieved by introducing a constraint to the dynamic equation of the single stance phase Equation (7.1):

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0. \tag{7.14}$$

This constraint keeps the swing foot on the ground while the model is in the double stance phase but since the ground can only push the robot (it can't pull), the vertical component of this constraint force should always be positive, i.e. $\lambda^{y}_{DS} > 0$. The constraint term should be added to equation of motion of single stance phase (Equation (7.1)) in order to obtain equation of motion of double stance phase:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{S}\mathbf{u} + \boldsymbol{\tau}\left(\mathbf{u}_{\mathrm{ba}}\right) + \mathbf{J}_c^{T}\lambda_{\mathrm{DS}}. \tag{7.15}$$

The constraint force λ_{DS} can be obtained by taking time derivative of equation (7.14) as:

$$\mathbf{J}_c\ddot{\mathbf{q}} + \dot{\mathbf{J}}_c\dot{\mathbf{q}} = 0, \tag{7.16}$$

and inserting it into (7.15) as follows:

$$\lambda_{\mathrm{DS}} = -(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^{T})^{-1}(\mathbf{J}_c\mathbf{M}^{-1}(\mathbf{S}\mathbf{u} + \boldsymbol{\tau}\left(\mathbf{u}_{\mathrm{ba}}\right) - \mathbf{C}\dot{\mathbf{q}} - \mathbf{G}) + \dot{\mathbf{J}}_c\dot{\mathbf{q}}). \tag{7.17}$$

7.2 Control

In this section, we will introduce to two controllers that we have used to achieve bipedal walking with variable stiffness biarticular springs. First controller is responsible for tracking the reference joint trajectories while the variable stiffness controller helps it overcome the deviations from trajetory. Figure 7.4 shows the controller diagram.

Reference joint trajectories and the variable stiffness controller is based on the varible stiffness spring-loaded inverted pendulum (VSLIP-SL) model [41].

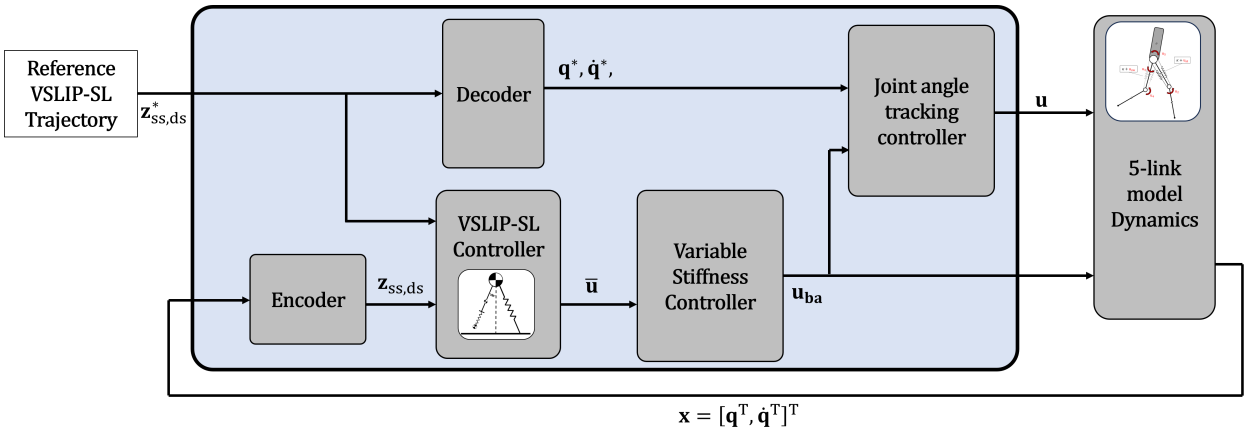


Figure 7.4: Controller diagram for the variable stiffness BA model

7.2.1 Trajectory Tracking Controller

In this section, proposed controllers will be introduced for the 5-link bipedal robot model with variable stiffness biarticular springs. Single stance phase controller uses the feedback linearization control to track the reference joint angles obtained via an encoder-decoder scheme similar to [41]. In the double stance phase, we've found that a simple PD controller is enough to achieve a satisfactory tracking results.

7.2.1.1 Single stance phase

We use feedback linearization to track actuated joint trajectories. Inputs can be chosen as:

$$\mathbf{u} = (\mathbf{T}\mathbf{M}^{-1}\mathbf{S})^{-1}(\boldsymbol{v} + \mathbf{T}\mathbf{M}^{-1}(\mathbf{H} - \boldsymbol{\tau}(\mathbf{u}_{\mathrm{ba}}))), \tag{7.18}$$

to linearize the system given in Equation (7.1) where $\mathbf{T} \in \mathbb{R}^{4\times 5}$ is the task space matrix that maps the generalized coordinates to the actuated ones and

$$\boldsymbol{v} = \mathrm{K_p}\boldsymbol{y} + \mathrm{K_d}\dot{\boldsymbol{y}}, \tag{7.19}$$

$$\boldsymbol{y} = \begin{bmatrix} \theta_2^*(\alpha) - \theta_2 \\ \theta_3^*(\alpha) - \theta_3 \\ \theta_4^*(\alpha) - \theta_4 \\ \theta_5^*(\alpha) - \theta_5 \end{bmatrix}. \tag{7.20}$$

$\mathrm{K_p}$ and $\mathrm{K_d}$ are the proportional and derivative gains and are set to same values for each actuated joint. Reference trajectories indicated by "*" superscript are obtained using the Decoder 7.2.2.

7.2.1.2 Double stance phase

For the double stance phase, we use a simple PD controller. Since this phase is relatively short, this controller works quite well.

$$\mathbf{u} = \mathbf{v}, \tag{7.21}$$

where $\mathbf{v}$ was given in Equation (7.19).

7.2.2 Encoder-Decoder

Now that we have defined the controllers for VSLIP-SL and the 5-link model, the remaining problem is to connect them together. We will achieve this through the reference trajectories indicated by * in Equation (7.20).

For the single stance phase, encoding will be done by solving the following for $\bar{\boldsymbol{q}} = [x_{\mathrm{M}}, y_{\mathrm{M}}, \theta, r]^T$:

$$\mathbf{F}_{\boldsymbol{x}}^{\mathrm{enc}} = \begin{bmatrix} x_{\mathrm{CoM}}(\boldsymbol{q}) - \bar{x}_{\mathrm{CoM}}(\bar{\boldsymbol{q}}) \\ y_{\mathrm{CoM}}(\boldsymbol{q}) - \bar{y}_{\mathrm{CoM}}(\bar{\boldsymbol{q}}) \\ x_{\mathrm{sw}}(\boldsymbol{q}) - \bar{x}_{\mathrm{sw}}(\bar{\boldsymbol{q}}) \\ y_{\mathrm{sw}}(\boldsymbol{q}) - \bar{y}_{\mathrm{sw}}(\bar{\boldsymbol{q}}) \end{bmatrix} = 0. \tag{7.22}$$

Then using the resulting $\bar{\boldsymbol{q}}$ to solve

$$\mathbf{F}_{\dot{\boldsymbol{x}}}^{\mathrm{enc}} = \begin{bmatrix} \dot{x}_{\mathrm{CoM}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\bar{x}}_{\mathrm{CoM}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \\ \dot{y}_{\mathrm{CoM}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\bar{y}}_{\mathrm{CoM}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \\ \dot{x}_{\mathrm{sw}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\bar{x}}_{\mathrm{sw}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \\ \dot{y}_{\mathrm{sw}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \dot{\bar{y}}_{\mathrm{sw}}(\bar{\boldsymbol{q}}, \dot{\bar{\boldsymbol{q}}}) \end{bmatrix} = 0, \tag{7.23}$$

for $\dot{\bar{\boldsymbol{q}}}$. Solving $\mathbf{F}_{\boldsymbol{x}}^{\mathrm{enc}}$ and $\mathbf{F}_{\dot{\boldsymbol{x}}}^{\mathrm{enc}}$ gives us the equivalent VSLIP-SL model to the current state of the 5-link model. Using $\bar{\boldsymbol{q}}$ and $\dot{\bar{\boldsymbol{q}}}$, we can get the response $\bar{\boldsymbol{u}}_{\mathrm{SS}}$ of the VSLIP-SL controller given by Equation (3.14).

For the double stance phase, we can directly use $x_{\mathrm{CoM}}(\boldsymbol{q})$, $y_{\mathrm{CoM}}(\boldsymbol{q})$, $\dot{x}_{\mathrm{CoM}}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, $\dot{y}_{\mathrm{CoM}}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ values of the 5-link model because these are already the states of VSLIP-SL in double stance phase. These can be used as inputs to the VSLIP-SL controller given in Equation (3.19) to obtain the $\bar{\boldsymbol{u}}_{\mathrm{DS}}$ values.

Next step is obtaining the desired joint trajectories using these values (decoding). By solving the sets of nonlinear equations

$$\mathbf{F}_{\boldsymbol{x}}^{\mathrm{dec}} = \begin{bmatrix} \bar{x}^*_{\mathrm{CoM}} - x_{\mathrm{CoM}}(\boldsymbol{q}^*) \\ \bar{y}^*_{\mathrm{CoM}} - y_{\mathrm{CoM}}(\boldsymbol{q}^*) \\ \bar{x}^*_{\mathrm{sw}} - x_{\mathrm{sw}}(\boldsymbol{q}^*) \\ \bar{y}^*_{\mathrm{sw}} - y_{\mathrm{sw}}(\boldsymbol{q}^*) \\ \theta^*_5 - \pi/2 \end{bmatrix} = 0, \tag{7.24}$$

$$\mathbf{F}_{\dot{\boldsymbol{x}}}^{\mathrm{dec}} = \begin{bmatrix} \dot{\bar{x}}^*_{\mathrm{CoM}} - \dot{x}_{\mathrm{CoM}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*) \\ \dot{\bar{y}}^*_{\mathrm{CoM}} - \dot{y}_{\mathrm{CoM}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*) \\ \dot{\bar{x}}^*_{\mathrm{sw}} - \dot{x}_{\mathrm{sw}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*) \\ \dot{\bar{y}}^*_{\mathrm{sw}} - \dot{y}_{\mathrm{sw}}(\boldsymbol{q}^*, \dot{\boldsymbol{q}}^*) \\ (\dot{\theta}_5^*)^2 \end{bmatrix} = 0, \tag{7.25}$$

consecutively, we can obtain the $\boldsymbol{q}^* = [\theta_1^*, \theta_2^*, \theta_3^*, \theta_4^*, \theta_5^*]^T$ and $\dot{\boldsymbol{q}}^*$ to be used in the joint angle tracking controller for the 5-link model (Equations (7.18) and (7.21)). In the double stance phase, $\bar{x}_{\text{sw}}^*$ and $\bar{y}_{\text{sw}}^*$ are set to the current position of the robot's foot on the ground.

We use the `fsolve` function of Matlab with the "Levenberg-Marquardt" option for solving sytems of nonlinear equations in this paper.

7.2.3 Variable Stiffness Controller

We can find the variable stiffness responses of the VSLIP-SL controller by using the outputs of the encoder $\bar{\boldsymbol{q}}$ and $\dot{\bar{\boldsymbol{q}}}$. The task of the variable stiffness controller of the 5-link biped robot is to convert these responses to stiffness changes of the stance and swing leg biarticular springs. This will be done based on the potential energy change of VSLIP-SL.

VSLIP-SL has nominal stiffness values k_{stLeg}, k_{swLeg}, k_{swFoot} and k_{DS}. Given the current state of the VSLIP-SL

$$\begin{gathered} \bar{\boldsymbol{q}}_{\text{SS}} = [x_{\text{M}}, y_{\text{M}}, \theta, r]^T \\ \text{or} \\ \bar{\boldsymbol{q}}_{\text{DS}} = [x_{\text{CoM}}, y_{\text{CoM}}]^T \end{gathered} \tag{7.26}$$

we can calculate the nominal potential energy stored in the springs. Nominal potential energies for single stance phase are:

$$\text{PE}_{\text{st, SS}}^{\text{nom.}} = \frac{1}{2}\text{k}_{\text{stLeg}}\Delta l_{\text{st, SS}}^2 \tag{7.27}$$

$$\text{PE}_{\text{sw, SS}}^{\text{nom.}} = \frac{1}{2}\text{k}_{\text{swLeg}}\Delta\theta^2 + \frac{1}{2}\text{k}_{\text{swFoot}}\Delta\text{r}^2 \tag{7.28}$$

where $\Delta l_{\text{st, SS}}$, $\Delta\theta$ and Δr are the deflections in the respective springs.

Nominal potential energies for double stance phase are:

$$\text{PE}_{\text{st, DS}}^{\text{nom.}} = \frac{1}{2}\text{k}_{\text{DS}}\Delta l_{\text{st, DS}}^2 \tag{7.29}$$

$$\text{PE}_{\text{sw, DS}}^{\text{nom.}} = \frac{1}{2}\text{k}_{\text{DS}}\Delta l_{\text{sw, DS}}^2 \tag{7.30}$$

where $\Delta l_{\text{sw, DS}}$ and $\Delta l_{\text{sw, DS}}$ are the deflections in the respective springs.

We can calculate the potential energy stored in the variable springs in the VSLIP-SL in a similar manner. Potential energies stored in the variable stiffness springs for single stance phase are:

$$\text{PE}_{\text{st, SS}}^{\text{var.}} = \frac{1}{2}(\text{k}_{\text{stLeg}} + \bar{\text{u}}_1)\Delta l_{\text{st, SS}}^2 \tag{7.31}$$

$$\mathrm{PE}^{\mathrm{var.}}_{\mathrm{sw,\,SS}} = \frac{1}{2}(\mathrm{k_{swLeg}} + \bar{\mathrm{u}}_2)\Delta\theta^2 + \frac{1}{2}(\mathrm{k_{swFoot}} + \bar{\mathrm{u}}_3)\Delta\mathrm{r}^2 \tag{7.32}$$

where $\Delta l_{\mathrm{st,\,SS}}$, $\Delta\theta$ and $\Delta\mathrm{r}$ are the deflections in the respective springs.

Nominal potential energies for double stance phase are:

$$\mathrm{PE}^{\mathrm{var.}}_{\mathrm{st,\,DS}} = \frac{1}{2}(\mathrm{k_{DS}} + \bar{\mathrm{u}}_4)\Delta l^2_{\mathrm{st,\,DS}} \tag{7.33}$$

$$\mathrm{PE}^{\mathrm{var.}}_{\mathrm{sw,\,DS}} = \frac{1}{2}(\mathrm{k_{DS}} + \bar{\mathrm{u}}_5)\Delta l^2_{\mathrm{sw,\,DS}} \tag{7.34}$$

where $\Delta l_{\mathrm{sw,\,DS}}$ and $\Delta l_{\mathrm{sw,\,DS}}$ are the deflections in the respective springs.

We can then obtain the ratio between the potential energy stored in the legs with nominal stiffnesses and variable stiffnesses:

$$p_i = \frac{\mathrm{PE}^{\mathrm{var.}}_i}{\mathrm{PE}^{\mathrm{nom.}}_i},\ i \in \{\mathrm{st}, \mathrm{sw}\}. \tag{7.35}$$

This gives an idea of the VSLIP-SL's control effort. $p > 1$ means that the respective springs became more stiff and $p < 1$ means the controller made them more compliant. We can use this ratio to adjust our biarticular spring stiffnesses by simply using:

$$\mathrm{u}_i = C\ p_i\ \kappa,\ i \in \{\mathrm{st}, \mathrm{sw}\} \tag{7.36}$$

where C is a "reduction" constant. We need this because VSLIP-SL can only respond to external pushed via its variable stiffness responses, whereas the biped model can respond via changing its biarticular spring stiffnesss values and by its actuators that input torque to the system. Because the 5-link model has additional actuators and controller, we reduce the variable stiffness response using this ratio. We set $C = 0.5$ in this study.

7.3 Simulation Results

We simulated walking using the proposed controller on MATLAB/Simulink. The model parameters are shown in Table 7.1. Controller gains were set as:

$$\mathrm{K_p} = 1000,\ \ \mathrm{K_d} = 75 \tag{7.37}$$

Table 7.1: 5 Link Model Parameters

$l_1 = l_4 : 0.48$ [m]	$l_2 = l_3 : 0.48$ [m]	$l_5 : 0.48$ [m]
$m_1 = m_4 : 5$ [kg]	$m_2 = m_3 : 5$ [kg]	$m_5 : 60$ [kg]
$I_i = m_i l_i^2/12$ [kg $\cdot$ m^2], $i = 1, 2, 3, 4, 5$		
$\kappa = 75$ [N/mm], $r = 2$		

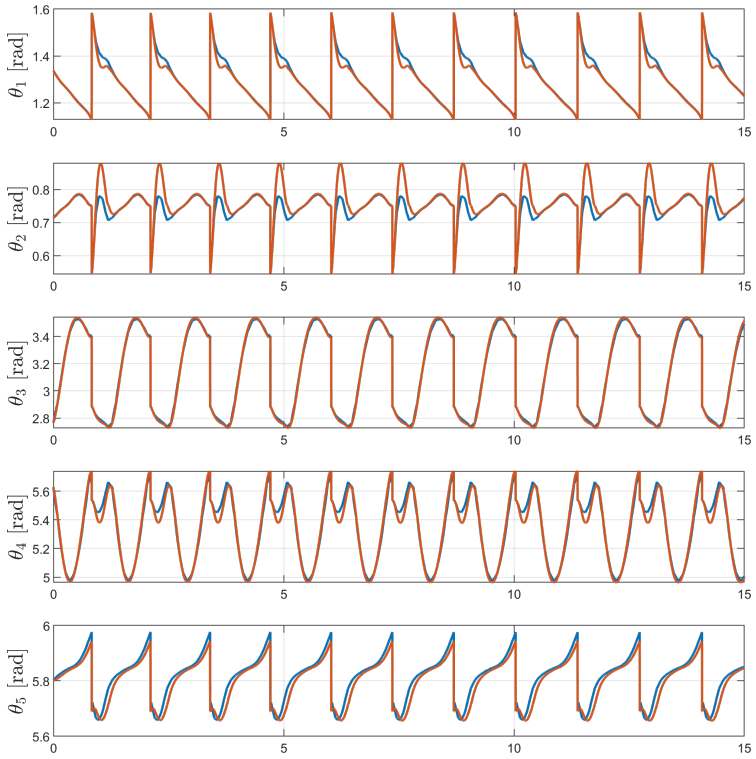


Figure 7.5: Trajectory tracking, no disturbance

Figure 7.5 shows the trajectory tracking results, Figure 7.6 input motor torques and Figure 7.7 how the controller adjusted the biarticular spring stiffnesses during the gait when there is no disturbance.

Then, we tried pushin the system with an impulse disturbance force of $F_{\text{dist.}} = [-100, 0]$ [N] at 10 second mark. Figure 7.5 shows the trajectory tracking results, Figure 7.6 input motor torques and Figure 7.7 how the controller adjusted the biarticular spring stiffnesses during the gait for the case with disturbance.

We can see that the controller can overcome the disturbance and keep the robot walking. Note that the walking fails when we disable the variable stiffness controller for $F_{\text{dist.}=[-100,0]}$ [N] at 10 second mark which shows the effectiveness of the controller.

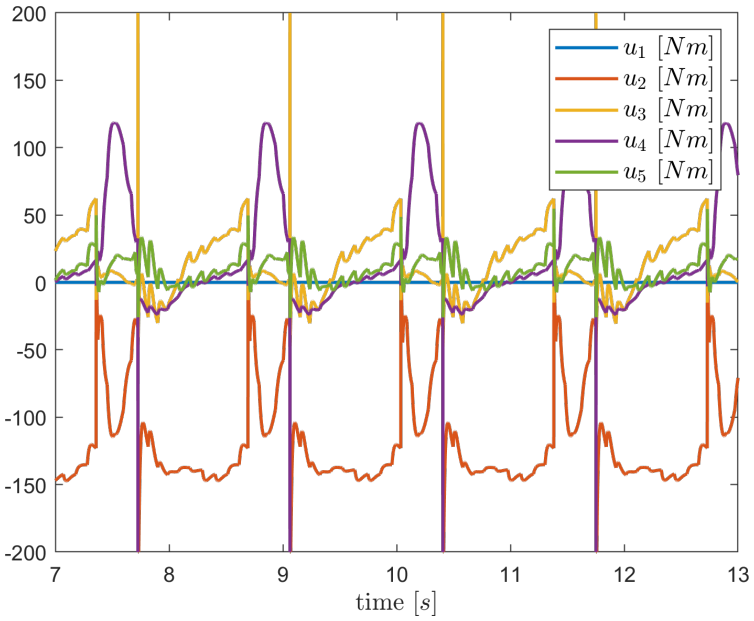


Figure 7.6: Motor torque inputs, no disturbance

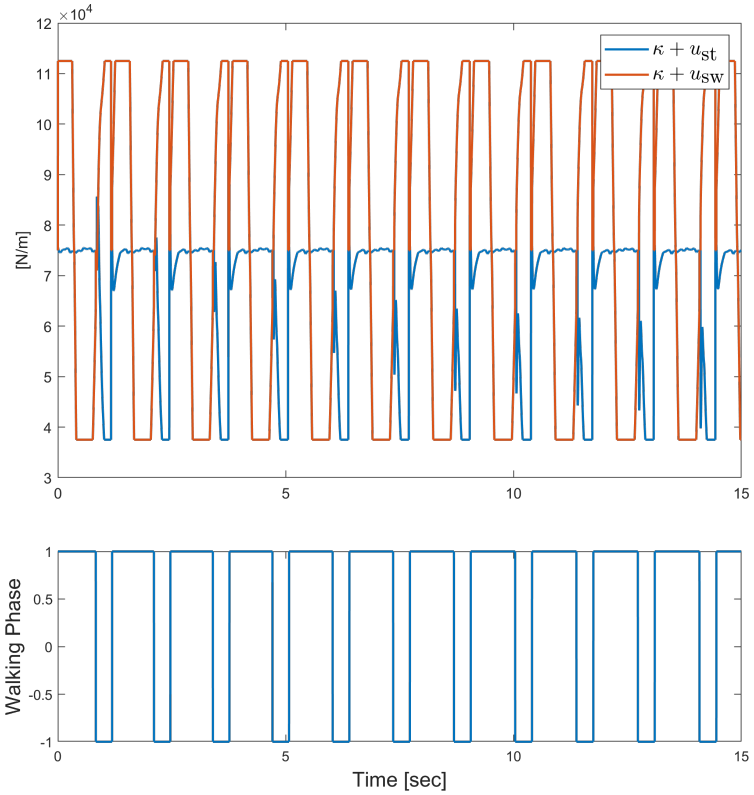


Figure 7.7: Biarticular muscle stiffness changes, no disturbance

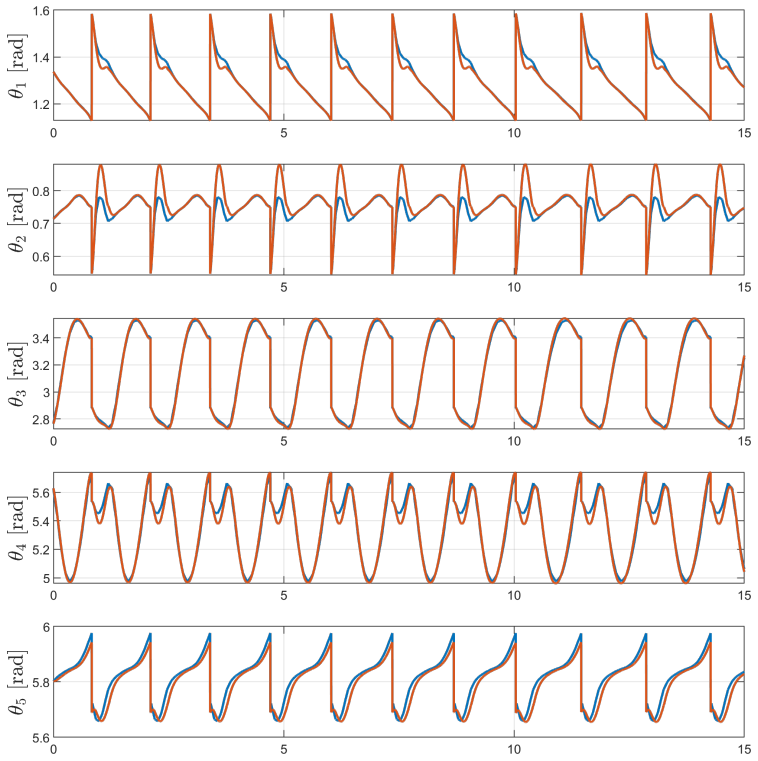


Figure 7.8: Trajectory tracking with $F_{\text{dist.}} = [-100, 0]$ [N] at 10 second mark disturbance

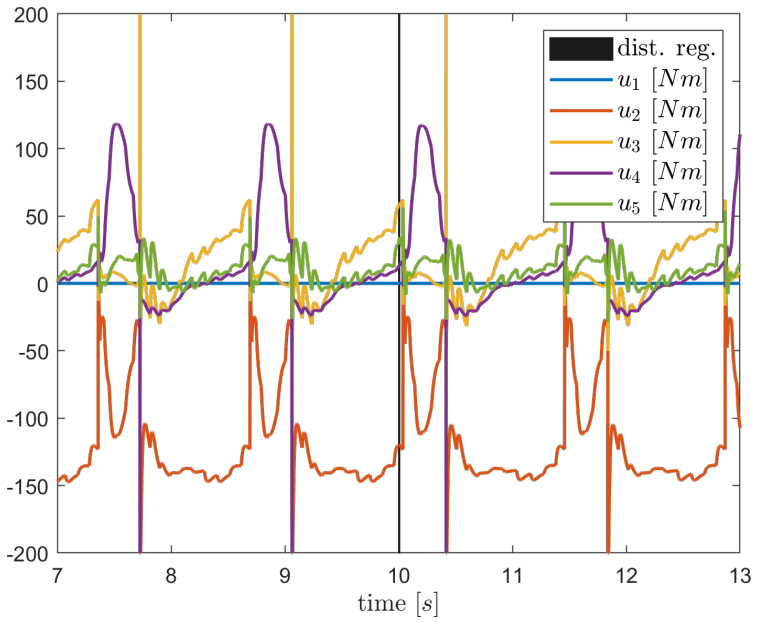


Figure 7.9: Motor torque inputs with $F_{\text{dist.}} = [-100, 0]$ [N] at 10 second mark

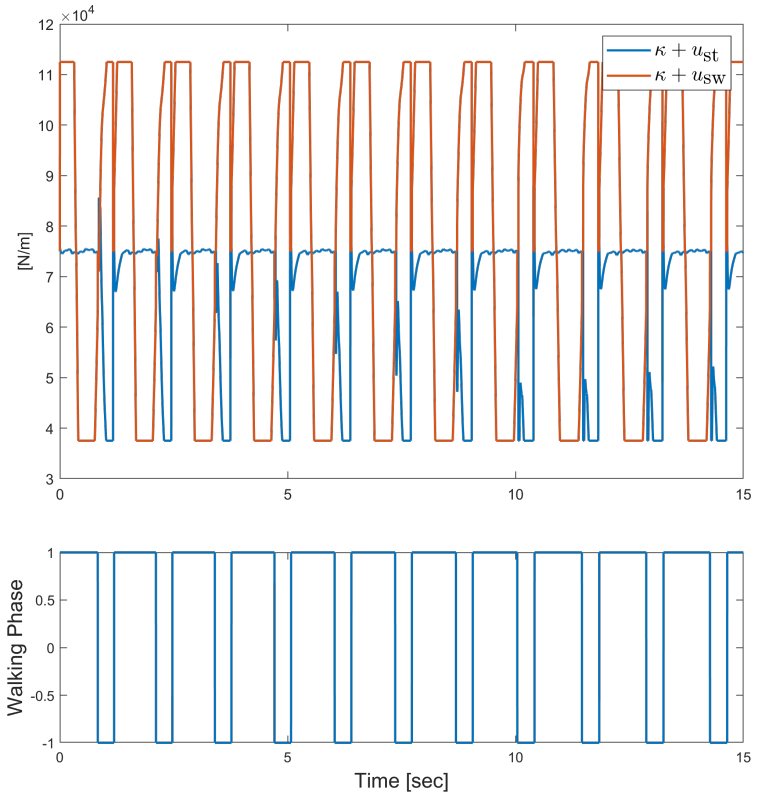


Figure 7.10: Biarticular muscle stiffness changes with $F_{\text{dist.}} = [-100, 0]$ [N] at 10 second mark

Chapter 8

Conclusion on Future Prospects

This work focused on improving the walking performance and robustness of biped walking robots. Various nonlinear control techniques, trajectory optimization techniques were employed throughout the thesis and their effectiveness was shown via simulations.

Spring-loaded inverted pendulum (SLIP) model is a well-known model in the field that can mimic the walking gait of an human. This simple model can be used as a base to achieve walking in bipedal robots but it lacks swing leg dynamics. In Chapter 2, we extend the SLIP model with swing leg dynamics while keeping its passive nature via our proposed spring-loaded inverted pendulum model with swing leg dynamics (SLIP-SL). We also show that this model can be used to control a 5-link fully actuated model via our propsed feedback linearization controller.

We further extend the SLIP-SL model by adding variable stiffness capability and an accompanying controller. This makes the template model robust against disturbances. Our idea is that if we make the template robust, the bipedal robot model that uses it in its controller will be more robust too. We then propose a controller that can embed the variable input responses in its actuator torques.

In Chapter 4, we investigate the effects of adding compliant elements in biarticular configuration to a 5-link bipedal robot model. We ran some trajectory optimization studies for various criteria. Results show that adding passive biarticular muscles improves the performance for all tested criteria, namely energy efficiency, maximum average velocity and in reducing motor torque requirements. We also propose a controller that can track the trajectories obtained in the optimization step. This controller can be used to reach a stable walking in a 5-link fully actuated bipedal robot model while having similar specific resistance values to the optimal trajectory.

In 5, we show that a active wobbling mass in torso when used in combination with the passive biarticular muscles can increase the maximum average walking speed of a 5-link point foot (underactuated) biped walking robot.

In 6, we propose method that enable a 5-link biped walker to walk on rough terrain without

any cameras or sensors that perceive the environment (terrain-blind walking). The proposed controller enables the biped robot to walk on terrains with up to 4.49 cm terrain height changes. We show that by adding biarticular muscles and tuning the stiffness and lever arm ratios properly, the height changes that the biped robot can handle is increased to 6.47 cm.

Finally, in Chapter 7 we assume a 5-link underactuated biped robot with variable stiffness biarticular springs. We propose a controller that can track the reference trajectories from the VSLIP-SL model while also mapping its variable stiffness responses to the biarticular spring stiffnesses. We show that the proposed controller can have the model reach a stable gait and can overcome external pushes.

As future work, we would like to investigate the effects of walking on soft-ground such as agricultural fields. We would like to see if the wobbling mass can improve terrain-blind walking performance. As this study usually investigated the steady-state walking gait, a robust and efficient control methods should be investigated where the robot can transition from standing-walking-running modes in a smooth manner. Finally, we would like the test the proposed methods on an real-life bipedal robot.

Bibliography

[1] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 2. IEEE, 2003, pp. 1620–1626.

[2] R. Takano and M. Yamakita, "Sequential-contact bipedal running based on slip model through zero moment point control," in *2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2017, pp. 1477–1482.

[3] H. Yano, J. Chang, R. Takano, and M. Yamakita, "Simultaneous optimization of trajectory and parameter for biped robot with series elastic actuators," *IFAC-PapersOnLine*, vol. 52, no. 22, pp. 7–12, 2019.

[4] A. Patel, S. L. Shield, S. Kazi, A. M. Johnson, and L. T. Biegler, "Contact-implicit trajectory optimization using orthogonal collocation," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2242–2249, 2019.

[5] J. Rummel, Y. Blum, and A. Seyfarth, "Robust and efficient walking with spring-like legs," *Bioinspiration & biomimetics*, vol. 5, no. 4, p. 046004, 2010.

[6] H. Geyer, A. Seyfarth, and R. Blickhan, "Compliant leg behaviour explains basic dynamics of walking and running," *Proceedings of the Royal Society B: Biological Sciences*, vol. 273, no. 1603, pp. 2861–2867, 2006.

[7] J. S. Park, C. M. Lee, S.-M. Koo, and C. H. Kim, "Gait phase detection using force sensing resistors," *IEEE Sensors Journal*, vol. 20, no. 12, pp. 6516–6523, 2020.

[8] M. A. Sharbafi, A. M. N. Rashty, C. Rode, and A. Seyfarth, "Reconstruction of human swing leg motion with passive biarticular muscle models," *Human movement science*, vol. 52, pp. 96–107, 2017.

[9] M. Kelly, "An introduction to trajectory optimization: How to do your own direct collocation," *SIAM Review*, vol. 59, no. 4, pp. 849–904, 2017.

[10] J. Koenemann, G. Licitra, M. Alp, and M. Diehl, "Openocl–open optimal control library," 2017.

[11] G. Garofalo, C. Ott, and A. Albu-Schäffer, "Walking control of fully actuated robots based on the bipedal slip model," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 1456–1463.

[12] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, vol. 4. IEEE, 1995, pp. 1942–1948.

[13] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback control of dynamic bipedal robot locomotion*. CRC press, 2018.

[14] R. Müller and R. Blickhan, "Running on uneven ground: leg adjustments to altered ground level," *Human movement science*, vol. 29, no. 4, pp. 578–589, 2010.

[15] L. Visser, S. Stramigioli, and R. Carloni, "Robust bipedal walking with variable leg stiffness," in *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. IEEE, 2012, pp. 1626–1631.

[16] W. Roozing, L. C. Visser, and R. Carloni, "Variable bipedal walking gait with variable leg stiffness," in *5th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics*. IEEE, 2014, pp. 931–938.

[17] J. Chang, M. M. Pelit, and M. Yamakita, "Slip-sl: Walking control based on an extended slip model with swing leg dynamics," *Advances in Science, Technology and Engineering Systems Journal*, vol. 6, no. 3, pp. 84–91, 2021.

[18] A. Isidori, *Nonlinear control systems: an introduction*. Springer, 1985.

[19] B. Fernini and M. Temmar, "The effect of mono and biarticular muscles on the dynamic of walking bipedal robot," in *2017 Intelligent Systems Conference (IntelliSys)*. IEEE, 2017, pp. 969–978.

[20] S. Oh, V. Salvucci, Y. Kimura, and Y. Hori, "Mathematical and experimental verification of efficient force transmission by biarticular muscle actuator," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 13 516–13 521, 2011.

[21] K. Hosoda, Y. Sakaguchi, H. Takayama, and T. Takuma, "Pneumatic-driven jumping robot with anthropomorphic muscular skeleton structure," *Autonomous Robots*, vol. 28, no. 3, pp. 307–316, 2010.

[22] D. Lakatos, C. Rode, A. Seyfarth, and A. Albu-Schäffer, "Design and control of compliantly actuated bipedal running robots: Concepts to exploit natural system dynamics," in *2014 IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2014, pp. 930–937.

[23] Y. Hanazawa, T. Hayashi, M. Yamakita, and F. Asano, "High-speed limit cycle walking for biped robots using active up-and-down motion control of wobbling mass," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 3649–3654.

[24] P. Meyns, S. M. Bruijn, and J. Duysens, "The how and why of arm swing during human walking," *Gait & posture*, vol. 38, no. 4, pp. 555–562, 2013.

[25] B. R. Umberger, "Effects of suppressing arm swing on kinematics, kinetics, and energetics of human walking," *Journal of biomechanics*, vol. 41, no. 11, pp. 2575–2580, 2008.

[26] L. C. Rome, L. Flynn, and T. D. Yoo, "Rubber bands reduce the cost of carrying loads," *Nature*, vol. 444, no. 7122, pp. 1023–1024, 2006.

[27] D. Tanaka, F. Asano, and I. Tokuda, "Gait analysis and efficiency improvement of passive dynamic walking of combined rimless wheel with wobbling mass," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 151–156.

[28] J. Chang, M. M. Pelit, H. Yano, H. Nishio, and M. Yamakita, "Verification of mechanical improvement for bipedal robot walking," *Transaction of the Japan Society for Simulation Technology*, vol. 14, no. 1, pp. 28–35, 2022.

[29] M. M. Pelit, J. Chang, and M. Yamakita, "Effects of passive biarticular muscles on walking performance for bipedal robots," in *2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2021, pp. 917–922.

[30] A. Werner, R. Lampariello, and C. Ott, "Trajectory optimization for walking robots with series elastic actuators," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 2964–2970.

[31] X. Da, O. Harib, R. Hartley, B. Griffin, and J. W. Grizzle, "From 2d design of underactuated bipedal gaits to 3d implementation: Walking with speed tracking," *IEEE Access*, vol. 4, pp. 3469–3478, 2016.

[32] M. S. Jones, "Optimal control of an underactuated bipedal robot," 2014.

[33] M. M. Pelit, J. Chang, R. Takano, and M. Yamakita, "Bipedal walking based on improved spring loaded inverted pendulum model with swing leg (slip-sl)," in *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2020, pp. 72–77.

[34] Y. Liu, P. M. Wensing, J. P. Schmiedeler, and D. E. Orin, "Terrain-blind humanoid walking based on a 3-d actuated dual-slip model," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 1073–1080, 2016.

[35] G. Zhao, O. Mohseni, M. Murcia, A. Seyfarth, and M. A. Sharbafi, "Exploring the effects of serial and parallel elasticity on a hopping robot," *Frontiers in Neurorobotics*, vol. 16, p. 919830, 2022.

[36] M. Kumamoto, T. Oshima, and T. Yamamoto, "Control properties induced by the existence of antagonistic pairs of bi-articular muscles—mechanical engineering model analyses," *Human Movement Science*, vol. 13, no. 5, pp. 611–634, 1994.

[37] J. Lee, G. Lee, S. Hong, S. Lee, J. H. Kim, and Y. Oh, "A novel multi-articular leg mechanism for biped robots inspired by bi-articular muscle," in *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. IEEE, 2016, pp. 1372–1377.

[38] G. Gabrielli, "What price speed? specific power required for propulsion of vehicles," *Mech. Eng.*, pp. 775–781, 1950.

[39] A. G. Rodríguez, J. Chacón, A. Donoso, and A. G. Rodríguez, "Design of an adjustable-stiffness spring: Mathematical modeling and simulation, fabrication and experimental validation," *Mechanism and Machine Theory*, vol. 46, no. 12, pp. 1970–1979, 2011.

[40] X. Xiong, X. Sun, W. Chen, Y. Zhi, and X. Fang, "Design of a variable stiffness actuator based on variable radius mechanisms," in *2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2022, pp. 1567–1572.

[41] M. M. Pelit and M. Yamakita, "Robust walking control based on the extended variable stiffness slip model," in *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 538–543.

[42] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, "An atlas of physical human–robot interaction," *Mechanism and Machine Theory*, vol. 43, no. 3, pp. 253–270, 2008.

[43] N. Hogan, "Adaptive control of mechanical impedance by coactivation of antagonist muscles," *IEEE Transactions on automatic control*, vol. 29, no. 8, pp. 681–690, 1984.

[44] E. Burdet, R. Osu, D. W. Franklin, T. E. Milner, and M. Kawato, "The central nervous system stabilizes unstable dynamics by learning optimal impedance," *Nature*, vol. 414, no. 6862, pp. 446–449, 2001.

[45] A. Jafari, N. G. Tsagarakis, and D. G. Caldwell, "A novel intrinsically energy efficient actuator with adjustable stiffness (awas)," *IEEE/ASME transactions on mechatronics*, vol. 18, no. 1, pp. 355–365, 2011.

[46] B. Ugurlu, C. Doppmann, M. Hamaya, P. Forni, T. Teramae, T. Noda, and J. Morimoto, "Variable ankle stiffness improves balance control: Experiments on a bipedal exoskeleton," *IEEE/ASME Transactions on mechatronics*, vol. 21, no. 1, pp. 79–87, 2015.

[47] M. M. Pelit and M. Yamakita, "Terrain-blind humanoid walking on rough terrain with trajectory optimization and biarticular springs," in *2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2023, p. to appear.