論文 / 著書情報
Article / Book Information

題目(和文)	
Title(English)	Realization of Robust and Efficient Bipedal Robot Mobility based on Mechanical Modular Devices and Control
著者(和文)	ジャンジュンホ
Author(English)	Junho Chang
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11926号, 授与年月日:2021年3月26日, 学位の種別:課程博士, 審査員:山北 昌毅,三平 満司,中島 求,畑中 健志,石崎 孝幸
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第11926号, Conferred date:2021/3/26, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

Realization of Robust and Efficient Bipedal Robot Mobility based on Mechanical Modular Devices and Control

by

Junho Chang

Dissertation for Degree of
Doctor of Engineering
Department of Systems and Control Engineering

Tokyo Institute of Technology
March 2021

東京工業大学
Tokyo Institute of Technology

Approved by:

Dr. Mitsuji Sampei
Professor
Department of Systems and Control Engineering
School of Engineering

Dr. Motomu Nakashima
Professor
Department of Systems and Control Engineering
School of Engineering

Dr. Takeshi Hatanaka
Associate Professor
Department of Systems and Control Engineering
School of Engineering

Dr. Takayuki Ishizaki
Associate Professor
Department of Systems and Control Engineering
School of Engineering

Dr. Masaki Yamakita
Professor [Academic Supervisor]
Department of Systems and Control Engineering
School of Engineering

Abstract

Robots in recent years play an important role in human society, such as improving human work efficiency and substituting dangerous tasks for humans. In particular, a legged mobile robot has been actively researched since it can adapt to human environment easily, and it has many advantages in mobility like it can move on rough terrain or change direction in narrow space.

Legged robots have been actively studied in terms of both mechanical and control aspects. A leg-type mobile robot consists of a relatively heavy upper body and a lower body to support the upper body, and the movement of a legged robot requires a high thrust to generate a force in the direction of travel while supporting the upper body with the legs, which causes a high ground reaction force. In addition, the leg-shaped robot loses kinetic energy due to the impact force of a high bandwidth at the moment when the foot touches the ground while moving. To overcome these phenomena, the mechanically devised methods such as attaching a series elastic element in the actuator or adding an inertial element in the joint of robot have been proposed. However, only a few studies for the control techniques that cover these mechanical improvements have been studied because of its complexity of dynamics, so realizing the mobility of the robot with these mechanical structures has been challengeable problem in the study of biped robots.

In the aspect of the control scheme, a control method for a legged mobile robot to realize movements such as walking and running has been actively researched for a few decades, and the feedback-controlled method is the most universally used to control for the robot, which contains generating the appropriate template gait for the target system

and calculating the control input based on stabilizing the whole states in periodic movement for the target system. This control method has several variations, such as a method of referencing a template model from the behavior of a simple system and a trajectory optimization method obtained by approximating the dynamics of a robot to a simple equation such as a polynomial.

When applying this method to robot walking or running, it is often better to modify each of the target systems to obtain better results. For instance, as a template model often used in walking or running control of a biped robot, the behavior of a simple system consisting of one spring and one point mass called the Spring Loaded Inverted Pendulum model (SLIP model) is widely used. This system is convenient because it is the simplest system that can simulate walking or running of robot, but there is a limit to expressing the dynamics of the swing leg or to applying directly to control for the robot model with a particular structure because of its simplicity. However, using a model that has been modified to suit each target system may result in better performance or easier control than conventional methods.

The purpose of this study is to improve the movement performance of a mechanically devised biped robot with a control method using simple template model dynamics. In this thesis, we present the following four contents that contribute to the purpose of this research. First, we present a new template model for biped walking robot which is modified version of the conventional SLIP model. Second, we present the new template model for 4-link walking robot to improve the robustness of walking. Third, we present a control method to use the template running gait for mechanically devised bipedal robot. Finally, we present a effectiveness verification of mechanically devised robot when the control scheme is based on the using a simple template model.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Robots in recent years play an important role in human society, such as improving human work efficiency and substituting dangerous tasks for humans. It has become possible to perform simple repetitive work in factories with an industrial manipulator, and to have robots help with medical-related surgery that requires precise work, and its application to human society is being promoted. Recently, research has been conducted to improve the applicability to a wide range of fields by autonomously moving robots with such diverse functions. In particular, a legged mobile robot has been actively researched since it can adapt to human environment easily because of its behavioral similarity to human's movement, and it has many advantages in mobility like it can move on rough terrain or change direction in narrow space.

Legged robots have been actively studied in terms of both mechanical and control aspects. A leg-type mobile robot consists of a relatively heavy upper body and a lower body to support the upper body, and the movement of a legged robot requires a high thrust to generate a force in the direction of travel while supporting the upper body with the legs, which causes a high ground reaction force. In addition, the leg-shaped robot loses kinetic energy due to the impact force of a high bandwidth at the moment when the foot touches the ground while moving. To overcome these phenomena, the mechanically devised methods such as attaching a series elastic element in the actuator or adding an inertial element in the joint of robot have been proposed. However, only a few studies for the control techniques that cover these mechanical improvements have been studied because of its complexity of dynamics, so realizing the mobility of the robot with these mechanical structures has been a challengeable problem in the study of biped robots.

1.1 Motivation

In the aspect of the control scheme, a control method for a legged mobile robot to realize movements such as walking and running has been actively researched for a few decades. A gait control method of legged robot in recent researches can be divided into several categories, such as feedback-control with template gaits and using a property of the passivity of the walking gait of robot. Feedback-control is the most universally used to control for the robot, which contains generating the appropriate template gaits for the target system and calculating the control input based on stabilizing the whole states in a periodic movement for the target system. This control method has several variations, such as a method of referencing a template model from the behavior of a simple system and a trajectory optimization method obtained by approximating the dynamics of a robot to a simple equation such as a polynomial.

When applying this method to robot walking or running, it is often better to modify each of the target systems to obtain better results. For instance, as a template model often used in walking or running control of a biped robot, the behavior of a simple system consisting of one spring and one point mass called the Spring Loaded Inverted Pendulum model (SLIP model) is widely used. This system is convenient because it is the simplest system that can simulate walking or running of robots, but there is a limit to expressing the dynamics of the swing leg or to applying directly to control for the robot model with a particular structure because of its simplicity. In this case, using a model that has been modified to suit each target system may result in better performance or easier control than conventional methods.

Another advantage of the method using a simple model is that the robustness of walking can be improved by giving an appropriate reference model [1]. Robustness of walking is an important problem in walking control of a biped robot because the stability of periodic walking is easily lost in the walking of a biped robot against disturbances such as steps and rough terrains. For instance, a limit cycle walking robot has been improved the energy efficiency of biped walking using the natural dynamics of the robot's movement, but have the problem of low robustness of the periodic walking [2]. This low robustness problem was relaxed by implementing the reference with the virtual model of a simple system. This result has been contributed to the potential for improved robustness by using a simple model as the controller reference, and the improvement of robustness by modifying this method could be expected.

From the view point of a structural idea, robots with the series elastic actuator (SEA) have been proposed. This mechanism is composed of a motor and a link and there exist elasticity elements between them. In the robotic researches, these mechanisms were adopted because of the adaptability for the unknown environment [3] or the energy storage for the robot's hopping [4]. They also can reduce the possibility of the damages of motors which are caused by high frequency impact forces by absorbing the impact forces with the elasticity. From these advantages, the SEA robots are expected to realize high-energy efficiency running gaits and the high-durability from the impact forces.

The purpose of this study is to improve the movement performance of mechanically devised biped robots with a control method using simple template model dynamics. In this thesis, we present the following four contents that contribute to the purpose of this research. As a simple template model used for walking control of robots, we proposed a new template model that considers the dynamics of the swing leg, which was not considered in the conventional research (Chapter 3). In order to clarify the advantages of the method of controlling using the template model, we confirmed that this method enhances walking performance and robustness (Chapter 4). We also proposed a control method for running a bipedal robot with a SEA mechanism (Chapter 5), and finally, we confirmed an improvement of the walking performance by devising SEA, inertia, and wobbling mass using the template model (Chapter 6).

1.2 Outline

The outline of this thesis consists of the following contents.

- Chapter 1 : Introduction

We present the background, motivation and purpose of this research.

- Chapter 2 : Preliminaries

In chapter 2, we describe the derivations of some mathematical tools used in this thesis. We describe the derivation of rigid body dynamics using Eular-Lagrange equation, derivation of constraint force from ground, and derivation of state changes when impact occurs. Then we describe some control schemes such as model matching control and output zeroing control.

- Chapter 3 : Bipedal Walking with Modified Simple Model

In chapter 3, we present the modified template model for bipedal walking robot. This modified model was developed to take swing leg dynamics into account in the Spring Loaded Pendulum model(SLIP model). The key to achieving periodic movement of this model is to modify the appropriate stiffness parameters, and we implemented this using a solver of a trajectory optimization problem. This periodic movement is generated under the condition that there is no input actuation for the template model, so that the result trajectory is only dependent on the natural dynamics of the simplified model. After calculating the template trajectory with appropriate parameters, we conducted the feedback control for multi-body bipedal robot with output zeroing method and verified the method with numerical simulation results.

- Chapter 4 : Robust Analysis of Biped Walking using Output Zeroing Control with Trajectory Optimization

In chapter 4, we present the robustness analysis of the bipedal walking using output zeroing control with a trajectory optimization. In the previous research, the robustness improvement of bipedal robot based on limit cycle walkers has been observed when they use the output zeroing control with the output function of relative degree 3, and we expand this result to the 4-link bipedal walking using the template dynamics generated by a trajectory optimization. Since the algorithm of optimizing trajectory is based on discretized dynamics, we used

Bézier curves to interpolate the optimized discrete states. We verified the result that the proposed method improves the robustness compared with the previous research, and confirmed the stable walking with the proposed method.

- Chapter 5 : Running Control of Bipedal Robot with Series Elastic Actuator

In chapter 5, we present a control method that realizes the running of bipedal robots consisting of series elastic actuators(SEA). In particular, we consider the SEA model which consists of a rotational link and a linear spring. Conventional control methods cannot be applied because of this mechanism, we proposed a systematic method to realize the stable running for the bipedal robot with series elasticity. This method uses the template dynamics model of non-SEA robot whose template dynamics is based on Spring Loaded Inverted Pendulum model(SLIP model). We confirmed the validity of the proposed method by numerical simulation results.

- Chapter 6 : Verification of Mechanical Improvement for Bipedal Walking Robot

In chapter 6, we present the verification of mechanical improvement for bipedal walking robots by adding series elasticity and mechanical elements to the joint structure including inerter and wobbling mass. We show the performance improvement in walking efficiency of the improved models compared to the nominal bipedal model.

- Chapter 7 : Conclusion

In chapter 7, we summarize the results of the thesis and present the future works for this research.

Chapter 2

Preliminaries

Since we realizes walking and running of robots using numerical simulation in this thesis, it is necessary to express the dynamics of robots used in each chapter from mathematical models. This chapter describes the derivation of mathematical models that express the dynamics of robots used in each chapter and a brief explanation of control methods.

2.1 Dynamics expression of biped system

The dynamics of walking and running of a biped robot are expressed by the rigid body dynamics that represent the behavior of the robot body, the constraint force on the robot's legs from the ground, and the state transition due to collision.

2.1.1 Derivation of rigid body dynamics motion

The dynamics of a rigid body can be expressed by several ways. In this thesis, we use the Euler-Lagrange equation for the expression of rigid body dynamics. There are several materials to derive this, but we will describe the simple method to derive the Euler-Lagrange equation based on the principle of least action.

Let us assume that the target robot system consists of N rigid bodies and there exist several constraints between them. First, we should define a generalized coordinate for this system to be satisfied the conditions of completeness and independence. Completeness is a condition for the system of the position to be uniquely determined, and independence is a condition to guarantee that there are no dependencies between position variables. In general case, the state of the system is not completely determined only by generalized coordinates, but also need the time derivative of the generalized coordinates, which is called generalized velocity.

After setting the generalized coordinates for the system, the next step is to define an action of the system. Suppose that two position of the system are given with coordinates as $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ at time $t = t_1$ and $t = t_2$, respectively. According to the principle of least action, the behavior of the system is uniquely determined for each state so as to minimize the action function determined by the generalized coordinates and velocity of the system where the initial position $\boldsymbol{q}_1$ and the final position $\boldsymbol{q}_2$ are fixed. The action function can be expressed by

$$S = \int_{t_1}^{t_2} \mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t)dt, \tag{2.1}$$

where S is the action of the system, and $\mathcal{L}$ is called Lagrangian of the system. Finding the conditions for the trajectory between $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ to be the minimum value of the equation(2.1), we use the variational method to find the stationary point of the states.

Suppose the $\boldsymbol{q}(t)$ is the function to make the action as minimal, and we consider the variation of the $\boldsymbol{q}(t)$ with $\boldsymbol{q}(t) + \delta\boldsymbol{q}(t)$. From our assumption that the initial position $\boldsymbol{q}_1 = \boldsymbol{q}(t_1)$ and the final position

$\boldsymbol{q}_2 = \boldsymbol{q}(t_2)$ are fixed, this gives the fact that

$$\delta \boldsymbol{q}(t_i) = 0, \qquad \forall i \in \{1, 2\}\,. \tag{2.2}$$

And, the variation of action δS can be written as

$$\begin{aligned}
\delta S &= \delta \left(\int_{t_1}^{t_2} \mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) dt \right) && (2.3)\\
&= \int_{t_1}^{t_2} \mathcal{L}(\boldsymbol{q} + \delta\boldsymbol{q}, \dot{\boldsymbol{q}} + \delta\dot{\boldsymbol{q}}, t) dt - \int_{t_1}^{t_2} \mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) dt && (2.4)\\
&= \int_{t_1}^{t_2} \left(\frac{\partial \mathcal{L}}{\partial \boldsymbol{q}} \delta\boldsymbol{q} + \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}} \delta\dot{\boldsymbol{q}} \right) dt && (2.5)\\
&= \int_{t_1}^{t_2} \left(\frac{\partial \mathcal{L}}{\partial \boldsymbol{q}} \delta\boldsymbol{q} + \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}} \delta\boldsymbol{q} \right) - \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}} \right) \delta\boldsymbol{q} \right) dt && (2.6)\\
&= \left. \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}} \delta\boldsymbol{q} \right|_{\boldsymbol{q}=\boldsymbol{q}_1}^{\boldsymbol{q}=\boldsymbol{q}_2} + \int_{t_1}^{t_2} \left(\frac{\partial \mathcal{L}}{\partial \boldsymbol{q}} - \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}} \right) \right) \delta\boldsymbol{q} dt. && (2.7)
\end{aligned}$$

The condition for stationary point is $\delta S = 0$, and using the condition of equation (2.2), the first term of equation (2.7) is equal to 0, and we can obtain

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}} \right) - \frac{\partial \mathcal{L}}{\partial \boldsymbol{q}} = 0. \tag{2.8}$$

Moreover, the Lagrangian in the rigid body dynamics is defined as

$$\mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) = \mathcal{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \mathcal{U}(\boldsymbol{q}), \tag{2.9}$$

where $\mathcal{T}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ and $\mathcal{U}(\boldsymbol{q})$ are the kinetic and potential energies and the kinetic energy is defined as

$$\mathcal{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}) := \frac{1}{2} \dot{\boldsymbol{q}}^T \boldsymbol{M}(\boldsymbol{q}) \dot{\boldsymbol{q}}. \tag{2.10}$$

Substituting the equation(2.9) to equation (2.8), we obtain

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}}}\right) - \frac{\partial \mathcal{L}}{\partial \boldsymbol{q}} = \frac{d}{dt}\left(\frac{\partial \mathcal{T}}{\partial \dot{\boldsymbol{q}}}\right) - \frac{\partial \mathcal{T}}{\partial \boldsymbol{q}} + \frac{\partial \mathcal{U}}{\partial \boldsymbol{q}} = 0. \tag{2.11}$$

And the first term of equation(2.11) can be expressed by substituting the equation(2.10)

$$\frac{d}{dt}\left(\frac{\partial \mathcal{T}}{\partial \dot{\boldsymbol{q}}}\right) = \frac{d}{dt}\left(\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}\right) = \dot{\boldsymbol{M}}\dot{\boldsymbol{q}} + \boldsymbol{M}\ddot{\boldsymbol{q}}. \tag{2.12}$$

So, the equation(2.11) can be written as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \left(\dot{\boldsymbol{M}}\dot{\boldsymbol{q}} - \frac{\partial \mathcal{T}}{\partial \boldsymbol{q}}\right) + \frac{\partial \mathcal{U}}{\partial \boldsymbol{q}} = 0. \tag{2.13}$$

Then, we define the second term as $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ and third term as $\boldsymbol{G}(\boldsymbol{q})$ in equation(2.13), we can get

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = 0, \tag{2.14}$$

as a dynamics of the rigid body motion.

2.1.2 Dynamics of constraint force

In the process of walking or running with a rigid system, there is a constraint such as a foot fixed on the ground and restraint to prevent joints from bending to the opposite side. This constraint makes the generalized coordinates of the system no longer independent, so that it no longer meets the requirements for deriving Euler-Lagrange's equation of motion. This problem can be solved by using the Lagrange multiplier method which will shown in the later. In this thesis, we will only deal with the holonomic constraints to describe the constraint of the biped robot, and details of this contents are stated in [5].

Holonomic constraints can be expressed using the generalized coordinates of the system as follows:

$$h_j(\boldsymbol{q}, t) = 0, \qquad j = 1, ..., N, \tag{2.15}$$

where $\boldsymbol{q} \in \mathbb{R}^n$ is the generalized coordinate of the n-dimensional system, and $N \in \mathbb{R}$ is the number of constraint conditions. Then, the constrained Lagrange equation can be derived with defining a new Lagrangian as

$$\mathcal{L}_c := \mathcal{L} - \sum_{j=1}^{N} \lambda_j h_j(\boldsymbol{q}, t), \tag{2.16}$$

where $\mathcal{L}$ is a Lagrangian of the system for an unconstrained condition. Using the calculus of variations with this Lagrangian, the stationary condition can be written as

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}_c}{\partial \dot{q}^i}\right) - \frac{\partial \mathcal{L}_c}{\partial q^i} = \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}^i}\right) - \frac{\partial \mathcal{L}}{\partial q^i} - \sum_{j=1}^{N} \lambda_j \frac{\partial h_j}{\partial q^i} = \tau^i, \tag{2.17}$$

where τ^i is defined as a generalized force for the unconstrained system. This condition gives the following dynamics equation of motion:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{\tau} + \boldsymbol{J}_c(\boldsymbol{q})^T \boldsymbol{\lambda}, \tag{2.18}$$

where $\boldsymbol{J}_c(\boldsymbol{q})$ is defined as

$$\boldsymbol{J}_c(\boldsymbol{q}) := \frac{\partial \boldsymbol{h}(\boldsymbol{q})}{\partial \boldsymbol{q}}. \tag{2.19}$$

Moreover, with the time derivative of the constraint equation, we can get

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0 \Rightarrow \boldsymbol{J}_c(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0. \tag{2.20}$$

Then, we can combine these equations as follow

$$\begin{bmatrix} \boldsymbol{M}(\boldsymbol{q}) & -\boldsymbol{J}_c^T \\ \boldsymbol{J}_c & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\tau} - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{G}(\boldsymbol{q}) \\ -\dot{\boldsymbol{J}}_c \dot{\boldsymbol{q}} \end{bmatrix}, \tag{2.21}$$

and multiplying the inverse matrix results in

$$\begin{bmatrix} \ddot{\boldsymbol{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{M}(\boldsymbol{q}) & -\boldsymbol{J}_c^T \\ \boldsymbol{J}_c & \boldsymbol{0} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{\tau} - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{G}(\boldsymbol{q}) \\ -\dot{\boldsymbol{J}}_c \dot{\boldsymbol{q}} \end{bmatrix}. \tag{2.22}$$

To calculate above equation, we use the formula of inverse of block matrix,

$$\begin{bmatrix} \boldsymbol{a} & \boldsymbol{b} \\ \boldsymbol{c} & \boldsymbol{d} \end{bmatrix}^{-1} = \begin{bmatrix} \boldsymbol{a}^{-1} + \boldsymbol{a}^{-1}\boldsymbol{b}\boldsymbol{s}^{-1}\boldsymbol{c}\boldsymbol{a}^{-1} & -\boldsymbol{a}^{-1}\boldsymbol{b}\boldsymbol{s}^{-1} \\ -\boldsymbol{s}^{-1}\boldsymbol{c}\boldsymbol{a}^{-1} & \boldsymbol{s}^{-1} \end{bmatrix} \tag{2.23}$$

where $\boldsymbol{s}$ is defined as $\boldsymbol{s} := \boldsymbol{d} - \boldsymbol{c}\boldsymbol{a}^{-1}\boldsymbol{b}$. Then, the constraint force $\boldsymbol{\lambda}$ can be expressed as

$$\boldsymbol{\lambda} = -\boldsymbol{X}(\boldsymbol{q})^{-1}(\boldsymbol{J}_c(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{\Gamma}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \dot{\boldsymbol{J}}_c(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}), \tag{2.24}$$

where,

$$\boldsymbol{X}(\boldsymbol{q}) = \boldsymbol{J}_c(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{J}_c(\boldsymbol{q})^T, \tag{2.25}$$

$$\boldsymbol{\Gamma}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{\tau} - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{G}(\boldsymbol{q}). \tag{2.26}$$

2.1.3 State transition by collision

When the biped robot touches the ground, there exists a discontinuous change of the system by collision with the ground. Since this phenomenon is unavoidable in the movement of the biped robot, the state transition due to collision is considered under the following assumptions.

- A collision is assumed to be inelastic.

- A collision is assumed to be instantaneous.

- There is no slippage during a collision.

Under these assumptions, we describe how to derive the state transition by collision as follows. Consider the following equation of motion,

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{\tau}. \tag{2.27}$$

At the instant of the collision, the dynamics can be written as follows,

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{\tau} + \delta\boldsymbol{F}, \tag{2.28}$$

where $\delta\boldsymbol{F}$ represents the impact force at collision. Integrating this equation from $t = 0^-$ to $t = 0^+$, this results in

$$\boldsymbol{M}(\boldsymbol{q})(\dot{\boldsymbol{q}}^+ - \dot{\boldsymbol{q}}^-) = \boldsymbol{F}, \tag{2.29}$$

where the Coriolis and centrifugal term, gravitational term and input term is assumed to be continuous in the moment of the collision. This impact force $\boldsymbol{F}$ can be expressed as the following

$$\boldsymbol{F} = \boldsymbol{J}_c(\boldsymbol{q})^T\boldsymbol{\lambda}, \tag{2.30}$$

where $\boldsymbol{J}_c(\boldsymbol{q})$ is the Jacobian matrix defined in (2.19). Moreover, the following equation is also satisfied in the moment of the collision.

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0. \tag{2.31}$$

Then, combining these equations, we can get

$$\begin{bmatrix} \boldsymbol{M}(\boldsymbol{q}) & -\boldsymbol{J}_c^T \\ \boldsymbol{J}_c & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{q}}^+ \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}^- \\ \boldsymbol{0} \end{bmatrix}. \tag{2.32}$$

Using the formula of (2.23), the transitioned state $\boldsymbol{q}^+$ can be expressed as

$$\dot{\boldsymbol{q}}^+ = (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{J}_c^T(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}\boldsymbol{J}_c)\dot{\boldsymbol{q}}^-. \tag{2.33}$$

2.2 Control scheme

In this section, we describe the basic method to control the bipedal robot. Since we are focus on the method that uses a template model, following two control methods will be treated in this thesis.

2.2.1 Output zeroing control

The output zeroing control is widely used for tracking the trajectory of the bipedal robot's gait. When this method used in the robotics field, the output function is usually chosen as the difference between the target trajectory and the state of the robot's joint angle, and by differentiating the output function, the dynamics of the output function can be expressed in terms of the state and the input of the system. By using this relationship, we can control the dynamics of the output function according to the purpose.

From the equation (2.14) in section 2.1.1, we derived the equation of motion for the rigid body dynamics. Combining this equation with a generalized force $\boldsymbol{\tau}$, this can be written as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{\tau}, \tag{2.34}$$

where $\boldsymbol{q} \in \mathbb{R}^n$, $\boldsymbol{M}(\boldsymbol{q}), \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$,$\boldsymbol{G}(\boldsymbol{q}) \in \mathbb{R}^n$, and $\boldsymbol{\tau} \in \mathbb{R}^n$. And if we suppose that the joints of the robot are actuated by m motors, the generalized force can be expressed as

$$\boldsymbol{\tau} = \boldsymbol{S}\boldsymbol{u}, \tag{2.35}$$

where $\boldsymbol{S} \in \mathbb{R}^{n\times m}$ is a transformation matrix and $\boldsymbol{u} \in \mathbb{R}^m$ is the motor torque. The equation (2.34) and (2.35) can be expressed using a state-space representation as

$$\begin{aligned}
\dot{\boldsymbol{x}} &= \begin{bmatrix} \dot{\boldsymbol{q}} \\ \ddot{\boldsymbol{q}} \end{bmatrix} && (2.36)\\
&= \begin{bmatrix} \dot{\boldsymbol{q}} \\ -\boldsymbol{M}(\boldsymbol{q})^{-1}\left(\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q})\right) \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{S} \end{bmatrix} \boldsymbol{u} && (2.37)\\
&= f(\boldsymbol{x}) + g(\boldsymbol{x})\boldsymbol{u} && (2.38)\\
\boldsymbol{y} &= h(\boldsymbol{x}), && (2.39)
\end{aligned}$$

where $\boldsymbol{x} := [\boldsymbol{q}^T, \dot{\boldsymbol{q}}^T]^T \in \mathbb{R}^{2n}$ is the state, and $\boldsymbol{y} \in \mathbb{R}^p$ is the output. For simplicity, we suppose the rank of the matrix $\boldsymbol{S}$ is m, and we treat the case of $p = m$. In robotics, the output function $\boldsymbol{y}$ is usually given as

$$\boldsymbol{y} = \boldsymbol{q} - \boldsymbol{q}_d, \tag{2.40}$$

where $\boldsymbol{q}_d$ is the goal trajectory of the joint angle.

And let us define the following expression

$$\begin{aligned}
L_f h(\boldsymbol{x}) &:= \sum_{i=1}^{n} \frac{\partial h}{\partial x_i} f_i(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) && (2.41)\\
L_f^i h(\boldsymbol{x}) &= L_f L_f^{i-1} h(\boldsymbol{x}) \quad \forall i \geq 2, && (2.42)
\end{aligned}$$

where f is a vector field defined on the manifold of the state, and h is a real-valued function. Using this expression, we describe the derivative of the output function $\boldsymbol{y}$ as

$$\begin{aligned}\frac{d\boldsymbol{y}}{dt} &= \frac{\partial h}{\partial \boldsymbol{x}}\dot{\boldsymbol{x}} = \frac{\partial h}{\partial \boldsymbol{x}}\left(f(\boldsymbol{x}) + g(\boldsymbol{x})\boldsymbol{u}\right) && (2.43)\\ &= L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}. && (2.44)\end{aligned}$$

If the input term appears in the first derivative of $\boldsymbol{y}$, $L_g h(\boldsymbol{x}) \neq 0$ is satisfied. But in the case that the input term appears in the kth derivative of $\boldsymbol{y}$,

$$\frac{d^k\boldsymbol{y}}{dt^k} = L_f^k h(\boldsymbol{x}) + L_g L_f^{k-1} h(\boldsymbol{x})\boldsymbol{u}, \tag{2.45}$$

where $L_h L_f^i h(\boldsymbol{x}) = 0, \quad \forall i \leq k-2$ is satisfied. Using this equation, we can stabilize the dynamics of $\boldsymbol{y}$. Suppose that the goal dynamics of the output function is given as

$$\boldsymbol{y}^{(k)} + \alpha_{k-1}\boldsymbol{y}^{(k-1)} + \cdots + \alpha_1 \boldsymbol{y} = 0, \tag{2.46}$$

so that $\boldsymbol{y}(t) \to 0$ as $t \to 0$. Then, the input for realizing this output function dynamics can be calculated as

$$\boldsymbol{u}^* = \left(L_g L_f^{k-1} h(\boldsymbol{x})\right)^{-1} \left(-L_f^k h(\boldsymbol{x}) - \alpha_{k-1}\boldsymbol{y}^{(k-1)} - \cdots - \alpha_1 \boldsymbol{y}\right). \tag{2.47}$$

2.2.2 Model matching control

Unlike the output zeroing control, the model matching control does not track the trajectory of the walking gait. Instead, this method calculates the input so that the target system mimics the dynamics of the template model. From this method, the behavior of the target system can inherit the dynamic characteristics of the template model.

Suppose that the dynamics of the biped robot is given as equation

(2.34) and (2.35), and the dynamics of the template model as

$$\tilde{\boldsymbol{M}}\ddot{\tilde{\boldsymbol{x}}} + \tilde{\boldsymbol{C}}\dot{\tilde{\boldsymbol{x}}} + \tilde{\boldsymbol{G}} = 0, \tag{2.48}$$

where $\boldsymbol{x} \in \mathbb{R}^r$ is the state of the template model. And let us assume that the state of the template model $\boldsymbol{x}$ can be written as a function of $\boldsymbol{q}$, i.e.,

$$\tilde{\boldsymbol{x}} = f(\boldsymbol{q}). \tag{2.49}$$

And, the derivative and the second derivative of this function gives

$$\dot{\tilde{\boldsymbol{x}}} = \boldsymbol{J}\dot{\boldsymbol{q}} \tag{2.50}$$

$$\ddot{\tilde{\boldsymbol{x}}} = \boldsymbol{J}\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}, \tag{2.51}$$

where $\boldsymbol{J} := \frac{\partial f}{\partial \boldsymbol{q}} \in \mathbb{R}^{r \times n}$ is the Jacobian matrix of the function f. Then, substituting equation (2.48) to the dynamics of the robot, it gives

$$\tilde{\boldsymbol{M}}\left(\boldsymbol{J}\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}\right) + \tilde{\boldsymbol{C}}\dot{\tilde{\boldsymbol{x}}} + \tilde{\boldsymbol{G}} = 0, \tag{2.52}$$

and from the equation (2.34) and (2.35),

$$\tilde{\boldsymbol{M}}\left(\boldsymbol{J}\boldsymbol{M}^{-1}\left(-\boldsymbol{C}\dot{\boldsymbol{q}} - \boldsymbol{G} + \boldsymbol{S}\boldsymbol{u}\right) + \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}\right) + \tilde{\boldsymbol{C}}\dot{\tilde{\boldsymbol{x}}} + \tilde{\boldsymbol{G}} = 0. \tag{2.53}$$

So we can calculate the input for mimicking the dynamics of the template model as

$$\boldsymbol{u} = \left(\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{S}\right)^{\dagger}\left(\boldsymbol{J}\boldsymbol{M}^{-1}\left(\boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{G}\right) - \dot{\boldsymbol{J}}\dot{\boldsymbol{q}} - \tilde{\boldsymbol{M}}^{-1}\left(\tilde{\boldsymbol{C}}\dot{\tilde{\boldsymbol{x}}} + \tilde{\boldsymbol{G}}\right)\right) \tag{2.54}$$

where $\boldsymbol{A}^{\dagger}$ means the pseudo-inverse matrix of $\boldsymbol{A}$.

Chapter 3

Bipedal walking with modified simple model

3.1 Introduction

Control methods for walking gait control have been developed for the last several decades, and there is a variety of methods to control the walking gait of a bipedal robot. A commonly used method for the biped robot is using the inverted pendulum model as a template in conjunction with the zero moment point(ZMP) [6], or optimizing the trajectories with optimization methods are often adopted for creating a template model [7]. Another popular method is to use a simple models that can recreate certain fundamental aspects of human or animal gait, as template models for bipedal robots. A template model that is commonly used for this purpose is called the bipedal spring loaded inverted pendulum (SLIP) model.

Many researches with this method have been proposed [8] [9] [10], however, the method using this model usually has one big assumption where the swing leg is assumed to move instantaneously to the proper touch-down position.This means that when the bipedal SLIP is chosen as the template to control the bipedal robot, the desired trajectory

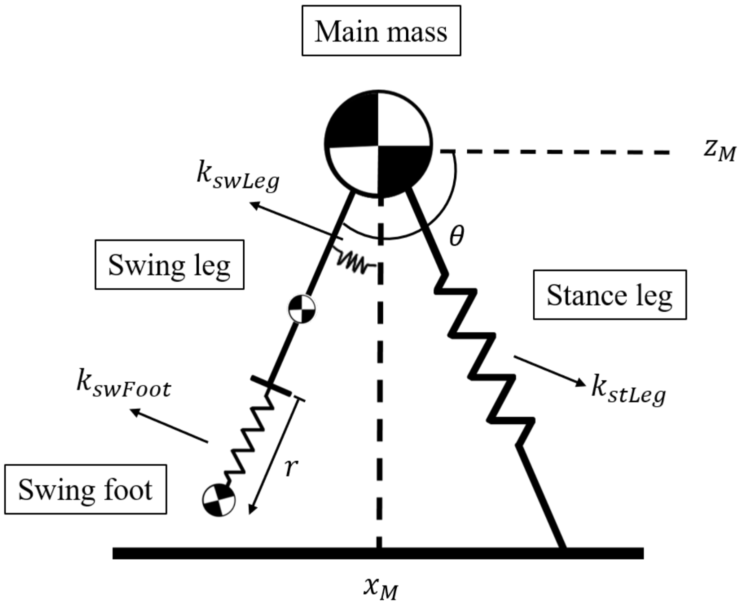


Figure 3.1: SLIP-SL model in single stance phase

for the swing foot can not be obtained from it. Additional steps are necessary to achieve the swinging motion. For example, [11] proposes a conceptual model where SLIP model is combined with a segmented leg so that the effects of swinging can be studied, but reference trajectories and a controller are necessary to achieve the desired motion.

In this chapter, an extended SLIP model will be proposed so that 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 (SLIP-SL) which is shown in Figure 3.1.

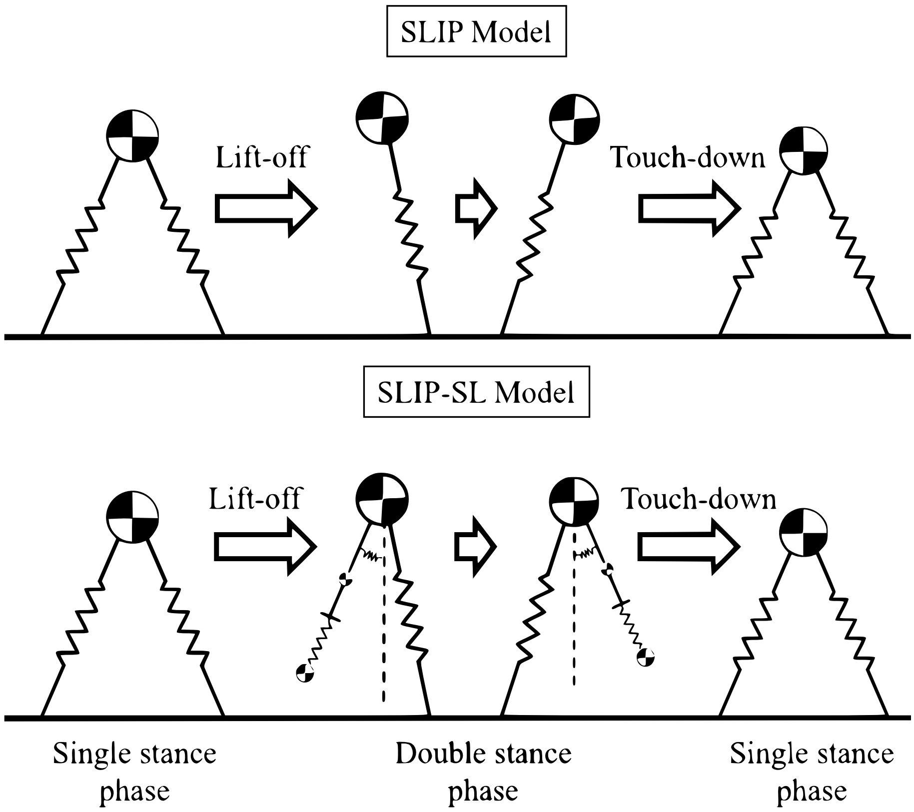


Figure 3.2: Bipedal SLIP model (top) and SLIP-SL model (bottom)

3.2 Dynamics model

This section represents the dynamics of a conventional bipedal SLIP model and the proposed model which is the improved version of the conventional bipedal SLIP model with swing leg dynamics.

3.2.1 Conventional bipedal SLIP model

The bipedal SLIP model consists of a mass and two linear springs, which can be seen in Figure 3.2. This point mass can be seen as the whole mass of the walker, and two massless springs are connected to this point mass. In the research of the conventional model [9], a periodic behavior of the system was shown as follows. This periodic behavior consists of two phases, single stance phase and double stance phase. Starting with the initial state of the single stance phase, this dynamics can be represented as the dynamics of one mass and one linear spring system with swing leg spring. It is usually assumed that the swing leg can instantly move to the desired touch down angle by reason of the fact that legs are assumed to be massless. The touch down event occurs when a point mass is near enough to the ground, then the phase of the system shifts to the double stance phase. By the reason that legs are to be massless, there is no effect of the impact event so that the whole physical energy of the system doesn't dissipate in the periodic behavior.In the double stance phase, the point mass is pushed by two springs until the one of the springs no longer can push it. When the point mass's position reaches to the natural length of the spring, the phase of the system again shifts to single stance phase from the double stance phase. They proved that this periodic movement can converge to a stable limit cycle under the conditions of the appropriate stiffness and initial conditions. However, from the above assumption that there is no inertial property in the swing leg, there is a limitation to consider the swing leg dynamics in the single stance phase with multi-body robots.

3.2.2 SLIP-SL model

Spring loaded inverted pendulum model with swing leg (SLIP-SL) can be seen in Figure 3.1 and its motion throughout a full step is represented in Figure 3.2.

To overcome the limitation of the conventional bipedal SLIP model, we propose a new template model for the bipedal walking, which is called Spring Loaded Inverted Pendulum with Swing Leg(SLIP-SL) model. This model has different dynamics models for each phase. In the double stance phase, this model has the same dynamics as the conventional bipedal SLIP model's one, so that the model consists of a point mass and two massless springs in the double stance phase. The big difference between the two is the addition of swing leg dynamics in the single stance phase. In the single stance phase, SLIP-SL consists of 3 massed elements which are the main mass, a massed swing leg and a point mass representing the swing foot and 3 springs which are the linear springs that connect the main mass to the ground, the torsional spring that is connected between the main point mass and the swing leg and the linear spring which connects the swing foot and the swing leg. After the touch-down event, SLIP-SL goes to the double stance phase, the swing elements disappear and the model becomes the same as the SLIP model in the respective phase.

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

$$\tilde{\boldsymbol{M}}(\tilde{\boldsymbol{q}})\ddot{\tilde{\boldsymbol{q}}} + \tilde{\boldsymbol{C}}(\tilde{\boldsymbol{q}}, \dot{\tilde{\boldsymbol{q}}}) + \tilde{\boldsymbol{G}}(\tilde{\boldsymbol{q}}) = \tilde{\boldsymbol{S}}\tilde{\boldsymbol{\tau}} \tag{3.1}$$

where the subscript $\tilde{\boldsymbol{q}} = [x_M, y_M, \theta, r]^T$ are the generalized coordinates, $\tilde{\boldsymbol{M}}(\tilde{\boldsymbol{q}}) \in \mathbb{R}^{4\times4}$ is an inertia matrix, $\tilde{\boldsymbol{C}}(\tilde{\boldsymbol{q}}, \dot{\tilde{\boldsymbol{q}}}) \in \mathbb{R}^{4\times4}$ is a Coriolis and centrifugal terms matrix, $\tilde{\boldsymbol{G}}(\tilde{\boldsymbol{q}}) \in \mathbb{R}^4$ is the gravity term and $\tilde{\boldsymbol{\tau}} \in \mathbb{R}^3$ 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. The resultant forces can be calculated as:

$$\tilde{\boldsymbol{\tau}} = \begin{bmatrix} k_{0,\mathrm{ss}}(L_{0,ss} - L_{st,ss}) \\ k_{\mathrm{swLeg}}(\theta_0 - \theta) \\ k_{\mathrm{swFoot}}(r_0 - r) \end{bmatrix}, \tag{3.2}$$

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 represent 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 vertical and horizontal 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 the swing foot point.

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

$$m \begin{bmatrix} \ddot{x}_{\mathrm{CoM}} \\ \ddot{y}_{\mathrm{CoM}} \end{bmatrix} = \boldsymbol{F}_{\mathrm{sw}} + \boldsymbol{F}_{\mathrm{st}} + m\boldsymbol{g}, \tag{3.3}$$

where $m_M, m_{\mathrm{swLeg}}, m_{\mathrm{swFoot}}$ represent the mass of the main body, the mass of the swing leg, and the mass of the swing feet in stance phase, respectively, and

$$m := m_M + m_{\mathrm{swLeg}} + m_{\mathrm{swFoot}}, \tag{3.4}$$

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:

$$\boldsymbol{F}_{st} = k_{0,\mathrm{ds}} \left(\frac{L_{0,\mathrm{ds}}}{L_{st}} - 1 \right) \left(\begin{bmatrix} x_{\mathrm{CoM}} \\ y_{\mathrm{CoM}} \end{bmatrix} - \begin{bmatrix} x_{foot} \\ 0 \end{bmatrix} \right), \tag{3.5}$$

$$\boldsymbol{F}_{sw} = k_{0,\mathrm{ds}} \left(\frac{L_{0,\mathrm{ds}}}{L_{sw}} - 1 \right) \left(\begin{bmatrix} x_{\mathrm{CoM}} \\ y_{\mathrm{CoM}} \end{bmatrix} - \begin{bmatrix} x^{-}_{foot} \\ 0 \end{bmatrix} \right), \tag{3.6}$$

where subscript “ds” indicates the double stance phase.

As can be seen, dynamics of SLIP-SL in the double stance phase are that of Bipedal SLIP model in its double stance phase.

3.2.3 Bipedal robot model

In this part, the dynamics of the 5 linked 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 assumed to be fully actuated and has an ankle torque.

Single stance phase

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

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{S}\boldsymbol{u}, \tag{3.7}$$

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 3.3 with the description of generalized 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. The single stance phase ends and the system goes into the double stance phase when the swing foot touches the ground.

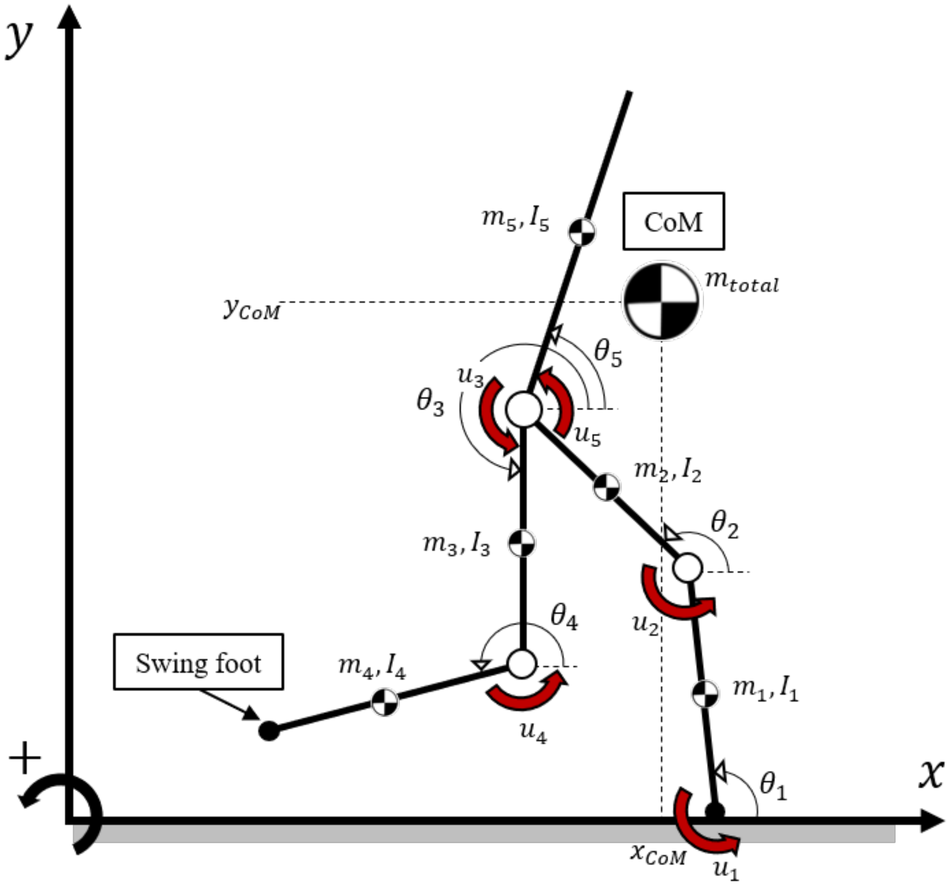


Figure 3.3: 5 link fully actuated robot model

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 the position is kept the same. This can be expressed as:

$$\boldsymbol{M}(\boldsymbol{q})\Delta\dot{\boldsymbol{q}} = \boldsymbol{J}_c^T \boldsymbol{\lambda}_{\text{impact}}, \tag{3.8}$$

where $\boldsymbol{J}_c \in \mathbb{R}^{2\times5}$ 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 $\boldsymbol{\lambda}_{\text{impact}} \in \mathbb{R}^2$. Assuming that the impact is inelastic, the velocity of the swing foot touching the ground will become zero after the impact which can be written as:

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}^+ = 0 \Leftrightarrow \boldsymbol{J}_c(\boldsymbol{q})\Delta\dot{\boldsymbol{q}} = -\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}}^-, \tag{3.9}$$

where $^-$ superscript indicates the moment just before the impact and $^+$ just after the impact. By solving (3.8) and (3.9) for $\boldsymbol{\lambda}_{impact}$ the following expression can be derived:

$$\boldsymbol{\lambda}_{\text{impact}} = -(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}\boldsymbol{J}_c\dot{\boldsymbol{q}}^-. \tag{3.10}$$

By inserting $\boldsymbol{\lambda}_{\text{impact}}$ into (3.8), we can obtain:

$$\dot{\boldsymbol{q}}^+ = (\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{J}_c^T(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}\boldsymbol{J}_c)\dot{\boldsymbol{q}}^-, \tag{3.11}$$

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).

Double stance phase

The system is now in the double stance phase where both feet are in contact with the ground. To model this, a constraint force 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:

$$\boldsymbol{J}_c(\boldsymbol{q})\dot{\boldsymbol{q}} = 0. \tag{3.12}$$

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

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{S}\boldsymbol{u} + \boldsymbol{J}_c^T \boldsymbol{\lambda}_{\text{ds}}. \tag{3.13}$$

The constraint force $\boldsymbol{\lambda}_{\text{ds}}$ can be obtained by taking a time derivative of equation (3.12) as:

$$\boldsymbol{J}_c\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}_c\dot{\boldsymbol{q}} = 0, \tag{3.14}$$

and inserting it into (3.13) as follows:

$$\boldsymbol{\lambda}_{\text{ds}}^{def} = -(\boldsymbol{J}_c\boldsymbol{M}^{-1}\boldsymbol{J}_c^T)^{-1}(\boldsymbol{J}_c\boldsymbol{M}^{-1}(\boldsymbol{S}\boldsymbol{u} - \boldsymbol{C}\dot{\boldsymbol{q}} - \boldsymbol{G}) + \dot{\boldsymbol{J}}_c\dot{\boldsymbol{q}}). \tag{3.15}$$

3.3 Trajectory optimization of bipedal robot

In this section, the optimization conducted to find periodic trajectories for SLIP-SL will be described. SLIP-SL is a passive model which means its gait solely depends on its mechanical parameters and initial conditions.In order to find the proper value, Direct Collocation Methods [12] were used in this study. These methods tackle the trajectory optimization problem by discretizing and converting it into a form which can be handled by nonlinear programming (NLP) solvers. There are many commercially available NLP solvers and in this study OpenOCL [13] will be used which can handle the multi-phase and simultaneous trajectory and parameter optimization problem. The optimization can be formulated as:

$$\begin{aligned} \min_{\boldsymbol{x}_i,\boldsymbol{p},T_i} \quad & \sum_{i=1}^{2}\left(\int_{T_{i-1}}^{T_i} J_i(\boldsymbol{x}_i(t),\boldsymbol{p})dt\right) \text{ for } i \in \{1,2\} \\ \text{s.t.} \quad & \dot{\boldsymbol{x}}_i = \boldsymbol{f}_i(\boldsymbol{x}_i(t),\boldsymbol{p}) \\ & \boldsymbol{r}_{i,k}(\boldsymbol{x}_i(\mu_{i,k}),\boldsymbol{p}) \leq 0, \end{aligned} \tag{3.16}$$

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, $\boldsymbol{f}_i(\boldsymbol{x},\boldsymbol{p})$ is the system dynamics function (differential equation) and $\boldsymbol{r}_{i,k}(\boldsymbol{x}_i,\boldsymbol{p})$ is the grid-constraints function. i represents the different stages of the optimization where $i = 1$ is representing the single stance phase and $i = 2$ representing the double stance phase of SLIP-SL (T_1 time point marks the touch-down event and T_2 marks the lift-off event). In this study, the Cost of Transport (CoT) [9] and a cost function to keep the swing foot low was used. CoT is an indicator of the walking efficiency.

Path, boundary and stage transition constraints are needed so that the solver can find feasible walking trajectories. In our study, path constraints are:

- Bounds were set for the parameters to be optimized:

$$\begin{cases} 15000 \leq k_{0,i} \leq 16000 \; [N/m], i \in \{ss, ds\} \\ 1 \leq L_{0,\text{ss}} \leq 1.2 \; [m] \\ 0 \leq k_{\text{swFoot}} \leq 20000 \; [N/m] \\ 0 \leq k_{\text{swLeg}} \leq 15000 \; [Nm/rad] \\ 0 \leq \theta_0 \leq 2\pi \; [rad] \\ -10 \leq r_0 \leq 10 \; [m] \end{cases} \tag{3.17}$$

- Stance leg spring in the single stance phase and both legs' springs in the double stance phase are always under contraction: $L_{st,ss} \leq L_{0,ss}$, $L_{st,ds} \leq L_{0,ds}$, $L_{sw,ds} \leq L_{0,ds}$

- Constraining the vertical position of CoM: $0\ [m] \leq y_{\text{CoM}} \leq 0.85\ [m]$

- Swing foot is always above the ground during the single stance phase: $y_{\text{swFoot}} \geq 0$

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

$$\left(\frac{x_{\text{swFoot}} - d_{obs}}{w_{obs}}\right)^2 + \left(\frac{y_{\text{swFoot}}}{h_{obs}}\right)^2 \geq 1, \tag{3.18}$$

  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: $\dot{x}_{\text{swFoot}} \geq 0$

- Vertical acceleration of CoM should be negative in the single stance phase so that system doesn't try to lift CoM up when there is only one leg on the ground. This is done to avoid large ankle torque when the SLIP-SL trajectory is used as a reference for the 5 linked model: $\ddot{y}_{\text{CoM},ss} \leq 0\ [m/s^2]$

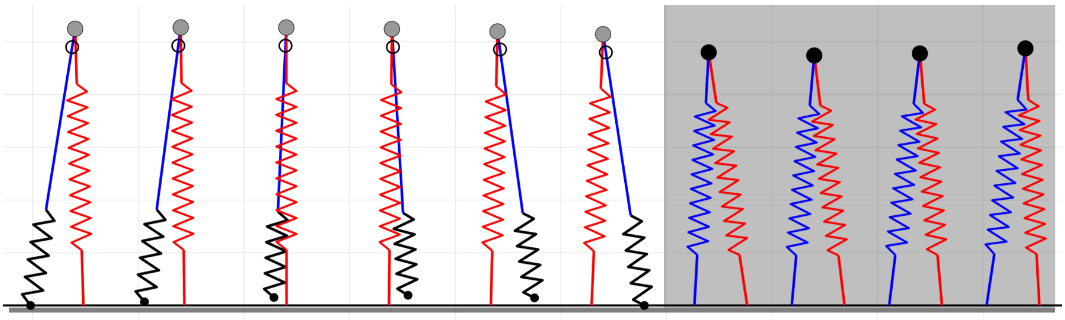


Figure 3.4: Snapshots of SLIP-SL's one step where gray dot in the single stance phase indicates the position of the point mass and the circle indicates the position of CoM

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: $y_{\text{swFoot}}(0) = 0$, $\dot{x}_{\text{swFoot}}(0) = 0$, $\dot{y}_{\text{swFoot}}(0) = 0$, $y_{\text{swFoot}}(T_1) = 0$
- Initial step length and the final step length should be same (for cyclic walking): $x_{\text{foot}} - x^{-}_{\text{foot}} = x^{+}_{\text{foot}} - x_{\text{foot}}$
- The initial position of the main mass relative to the stance foot should be the same as the final one (for cyclic walking): $x_{stFoot,0} - x_M(0) = x_{stFoot}(T_2) - x_M(T_2)$
- Constraints for cyclic walking: $y_{\text{CoM}}(0) = y_{\text{CoM}}(T_2)$, $\dot{x}_{\text{CoM}}(0) = \dot{x}_{\text{CoM}}(T_2)$, $\dot{y}_{\text{CoM}}(0) = \dot{y}_{\text{CoM}}(T_2)$
- 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: $L_{\text{sw,ds}}(T_2) = L_{0,\text{ds}}$

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 3.5 for various trajectories and a snapshot of SLIP-SL's one step can be seen in Figure 3.4 for a sample trajectory. Trajectory 'A' from Figure 3.5 will be used as the reference in Section 3.5 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 3.1.

Table 3.1: SLIP-SL's constant mechanical parameters

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

3.4 Control scheme

In this section, a 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 method, in a similar manner to [14] where a total energy control approach was used with the bipedal SLIP model. However, in this study, a trajectory tracking approach will be used.

3.4.1 Single stance phase

For the control of the robot in the single stance phase, there are three main tasks:

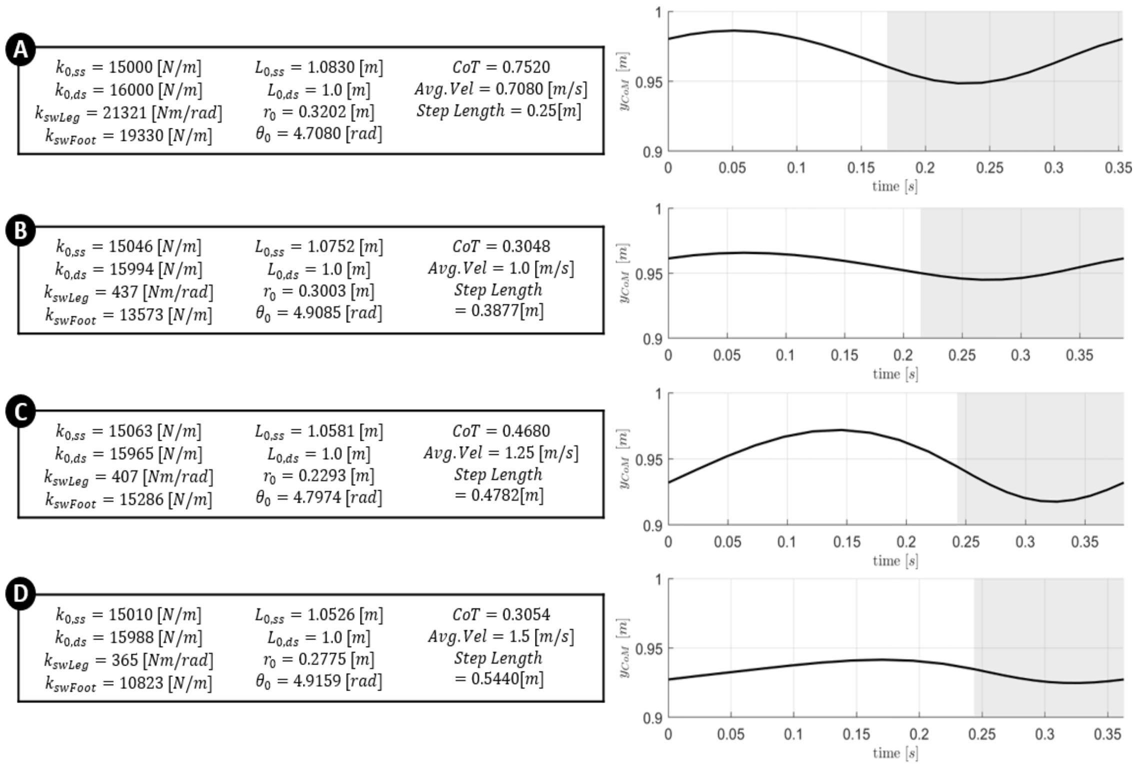


Figure 3.5: Optimization results for various SLIP-SL trajectories. For the trajectory A, step length was constrained to 0.25 [m] to get better ankle torques and this trajectory is the one that was used as reference in Section 3.4. For B, C and D trajectories, average velocity constraints were added. 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.

- 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:

$$\dot{\boldsymbol{x}}_{t,\mathrm{ss}} = \boldsymbol{J}_{t,\mathrm{ss}}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{3.19}$$

where

$$\dot{\boldsymbol{x}}_{t,\mathrm{ss}} = \begin{bmatrix} \dot{\boldsymbol{x}}_G \\ \dot{\boldsymbol{\xi}} \\ \dot{\theta}_5 \end{bmatrix}, \tag{3.20}$$

which is the velocity in the task space where subscript "ss" indicates the single stance phase and

$$\boldsymbol{J}_{t,ss}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{J}_G \\ \boldsymbol{J}_\xi \\ \boldsymbol{J}_{\theta_5} \end{bmatrix}, \tag{3.21}$$

are the combination Jacobian matrices. $\boldsymbol{J}_G$ maps generalized velocities to 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 trunk's angular velocity. By taking the time derivative of Equation (3.19) and inserting the obtained $\ddot{\boldsymbol{q}}$ into Equation (3.7) we can get:

$$\ddot{\boldsymbol{x}}_t = \boldsymbol{J}_{t,\mathrm{ss}}\boldsymbol{M}^{-1}(\boldsymbol{S}\boldsymbol{u} - \boldsymbol{C}\dot{\boldsymbol{q}} - \boldsymbol{G}) + \dot{\boldsymbol{J}}_{t,\mathrm{ss}}\dot{\boldsymbol{q}}. \tag{3.22}$$

Inputs should be chosen as:

$$\boldsymbol{u} = \boldsymbol{S}^{-1}\left(\boldsymbol{M}\boldsymbol{J}_{t,\mathrm{ss}}^{-1}(\ddot{\boldsymbol{x}}_{t_d,ss} - \dot{\boldsymbol{J}}_{t,\mathrm{ss}}\dot{\boldsymbol{q}}) + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{G}\right), \tag{3.23}$$

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

$$\ddot{\boldsymbol{x}}_{t_d,\mathrm{ss}} = \begin{bmatrix} K_{P_G}(x_{\mathrm{CoM,des}} - x_{\mathrm{CoM}}) + K_{D_G}(\dot{x}_{\mathrm{CoM,des}} - \dot{x}_{\mathrm{CoM}}) \\ K_{P_G}(y_{\mathrm{CoM,des}} - y_{\mathrm{CoM}}) + K_{D_G}(\dot{y}_{\mathrm{CoM,des}} - \dot{y}_{\mathrm{CoM}}) \\ K_{P_{\mathrm{sw}}}(x_{\mathrm{sw,des}} - x_{\mathrm{sw}}) + K_{D_{\mathrm{sw}}}(\dot{x}_{\mathrm{sw,des}} - \dot{x}_{\mathrm{sw}}) \\ K_{P_{\mathrm{sw}}}(y_{\mathrm{sw,des}} - y_{\mathrm{sw}}) + K_{D_{\mathrm{sw}}}(\dot{y}_{\mathrm{sw,des}} - \dot{y}_{\mathrm{sw}}) \\ K_{P_T}(\theta_{5,\mathrm{des}} - \theta_5) + K_{D_T}(-\dot{\theta}_5) \end{bmatrix} \tag{3.24}$$

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 3.3. In this study, the desired trunk angle was chosen as $\theta_{5,\mathrm{des}} = \pi$ [rad].

3.4.2 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 than becomes:

$$\dot{\boldsymbol{x}}_{t,ds} = \begin{bmatrix} \dot{\boldsymbol{x}}_G \\ \dot{\boldsymbol{\theta}}_5 \end{bmatrix} \in \mathbb{R}^3. \tag{3.25}$$

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:

$$\dot{\boldsymbol{q}}_{\text{hip}} = \boldsymbol{J}_{\text{hip,sw}}(\boldsymbol{q}_{\text{sw}})\dot{\boldsymbol{q}}_{\text{sw}} = \boldsymbol{J}_{\text{hip,st}}(\boldsymbol{q}_{\text{st}})\dot{\boldsymbol{q}}_{\text{st}}, \tag{3.26}$$

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 (3.26),

$$\dot{\boldsymbol{q}}_{\text{st}} = \boldsymbol{J}^{-1}_{\text{hip,st}}(\boldsymbol{q}_{\text{st}})\boldsymbol{J}_{\text{hip,sw}}(\boldsymbol{q}_{\text{sw}})\dot{\boldsymbol{q}}_{\text{sw}}, \tag{3.27}$$

can be obtained to get:

$$\dot{\boldsymbol{q}} = \underbrace{\begin{bmatrix} \boldsymbol{J}^{-1}_{\text{hip,st}}\boldsymbol{J}_{\text{hip,sw}} & 0_{2\times1} \\ I_{2\times2} & 0_{2\times1} \\ 0_{1\times2} & 1 \end{bmatrix}}_{\boldsymbol{\Gamma}(\boldsymbol{q})} \underbrace{\begin{bmatrix} \dot{\boldsymbol{q}}_{\text{sw}} \\ \dot{\theta}_5 \end{bmatrix}}_{\dot{\boldsymbol{q}}_a}. \tag{3.28}$$

After taking the time derivative of (3.28), then substituting the $\ddot{\boldsymbol{q}}$ and $\dot{\boldsymbol{q}}$ terms from (3.28) into (3.7) and multiplying with $\boldsymbol{\Gamma}^T$ from left, following can be obtained:

$$\boldsymbol{M}_a(\boldsymbol{q})\ddot{\boldsymbol{q}}_a + \boldsymbol{C}_a(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}_a + \boldsymbol{G}_a(\boldsymbol{q}) = \boldsymbol{u}_a, \tag{3.29}$$

where

$$\begin{cases} \boldsymbol{M}_a = \boldsymbol{\Gamma}^T\boldsymbol{M}\boldsymbol{\Gamma} \\ \boldsymbol{C}_a = \boldsymbol{\Gamma}^T(\boldsymbol{C}\boldsymbol{\Gamma} + \boldsymbol{M}\dot{\boldsymbol{\Gamma}}) \\ \boldsymbol{G}_a = \boldsymbol{\Gamma}^T\boldsymbol{G} \\ \boldsymbol{u}_a = \boldsymbol{\Gamma}^T\boldsymbol{u}. \end{cases} \tag{3.30}$$

Using a similar technique that was used for deriving (3.23):

$$\boldsymbol{u}_a = \boldsymbol{M}_a\boldsymbol{J}^{-1}_{t,\text{ds}}(\ddot{\boldsymbol{x}}_{t_d,\text{ds}} - \dot{\boldsymbol{J}}_{t,\text{ds}}\dot{\boldsymbol{q}}_a) + \boldsymbol{C}_a\dot{\boldsymbol{q}}_a + \boldsymbol{G}_a, \tag{3.31}$$

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

$$\ddot{\boldsymbol{x}}_{t_d,\mathrm{ds}} = \begin{bmatrix} K_{P_G}(x_{\mathrm{CoM,\ des}} - x_{\mathrm{CoM}}) + K_{D_G}(\dot{x}_{\mathrm{CoM}} - \dot{x}_{\mathrm{CoM}}) \\ K_{P_G}(y_{\mathrm{CoM,\ des}} - y_{\mathrm{CoM}}) + K_{D_G}(\dot{y}_{\mathrm{CoM,\ slipsl}} - \dot{y}_{\mathrm{CoM}}) \\ K_{P_T}(\theta_{5,\mathrm{des}} - \theta_5) + K_{D_T}(-\dot{\theta}_5) \end{bmatrix}. \quad (3.32)$$

However, by this way only $\boldsymbol{u}_a \in \mathbb{R}^3$ can be obtained which only has dimension 3 where $\boldsymbol{u} \in \mathbb{R}^5$ was needed. One way to calculate $\boldsymbol{u}$ is by using the relation $\boldsymbol{u}_a = \boldsymbol{\Gamma}^T \boldsymbol{u}$ given in (3.30) by:

$$\boldsymbol{u} = \boldsymbol{S}^{-1} \left(\underbrace{\boldsymbol{W}^{-1}\boldsymbol{\Gamma}(\boldsymbol{\Gamma}^T\boldsymbol{W}^{-1}\boldsymbol{\Gamma})^{-1}}_{(\boldsymbol{\Gamma}^T)^{+\boldsymbol{W}}} \boldsymbol{u}_a \right), \quad (3.33)$$

where $(\boldsymbol{\Gamma}^T)^{+\boldsymbol{W}}$ is the weighted matrix inverse operation. $\boldsymbol{W}$ is a positive definite matrix which is used to penalize the high ankle torques.

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 the single stance phase, the 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.

3.5 Simulation results and discussion

In the following simulation study, the designed controllers ability to track the reference SLIP-SL trajectories and the suitability of the SLIP-SL as a template model for walking will be tested. Table 3.2 shows the mechanical parameters of the 5 link robot and Table 3.3 shows the chosen gain values for the controller and the torque limits on the motors were set at 200 $[Nm]$.

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 3.6 shows the CoM trajectory and trunk orientation of the controlled system with the proposed controller. Figure 3.8 shows the swing foot trajectories 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 3.3. The reference SLIP-SL trajectories for the swing foot would not be available a template such as the popular SLIP was used.

Table 3.2: 5 link model parameters

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

Table 3.3: Control parameters

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

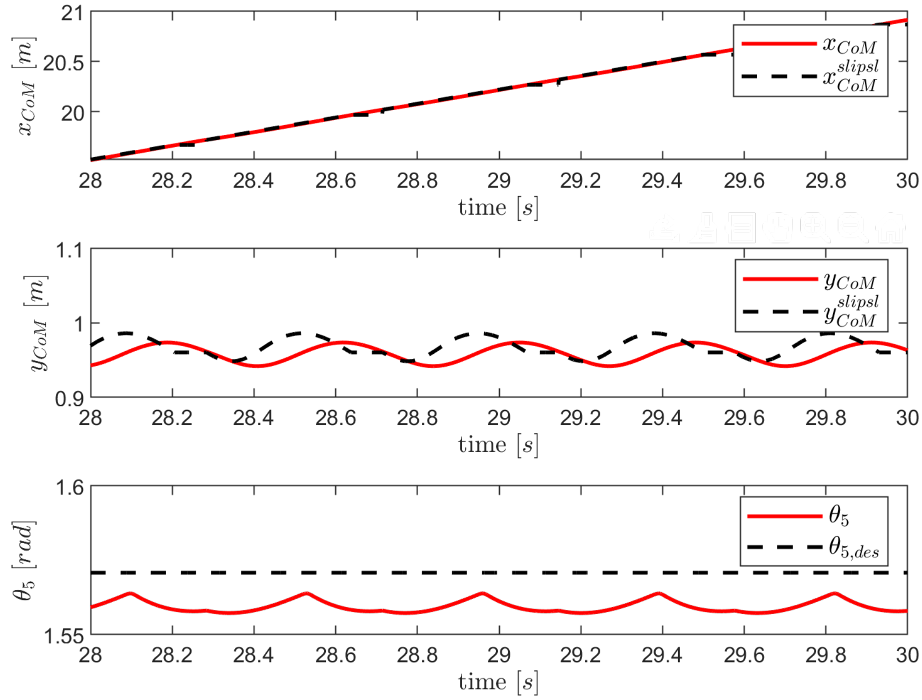


Figure 3.6: Trajectory tracking results for CoM horizontal position, vertical position and trunk orientation

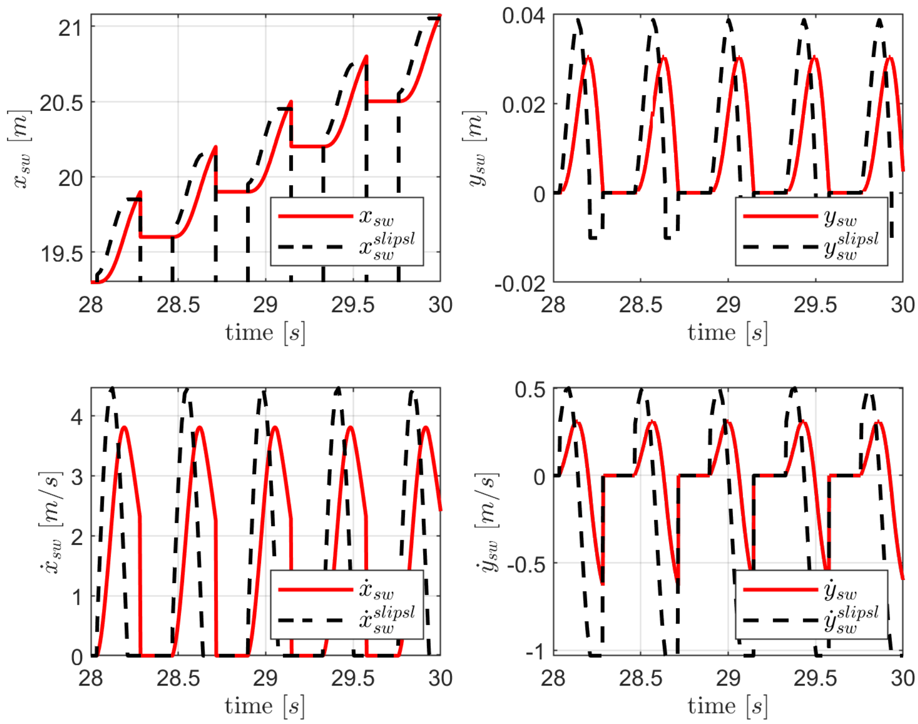


Figure 3.7: Trajectory tracking results for the swing foot

It can be seen in Table 3.3 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 [15]. For this trajectory, center of pressure stays within a 30 [cm] foot.

The figure shows snapshots of one stop from the 5 link models gait. to check the stability of the gait, Poincaré map approach was considered [14]. 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 because 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 [14]. Poincare Map for the controlled 5 link model is shown in Figure 3.9. It can be seen that the gait converges to a stable point in the section after a couple of steps which indicates stability.

In this study, it was also shown that by using the direct collocation optimization, various SLIP-SL trajectories can be obtained (Figure 3.5) that resemble the walking gaits. 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.

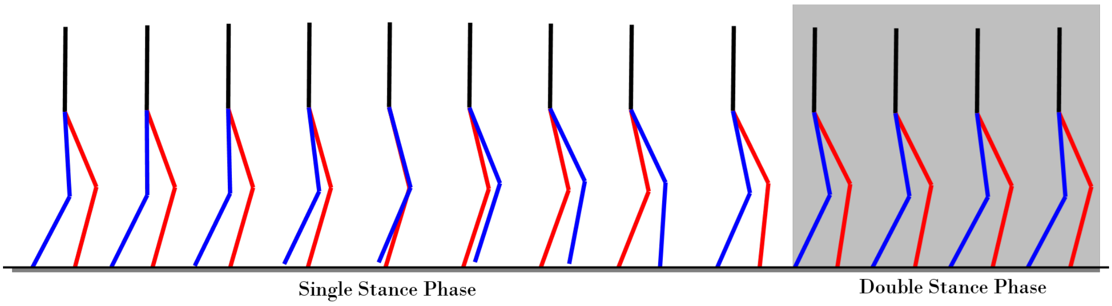


Figure 3.8: Snapshots of 5 Link models one step

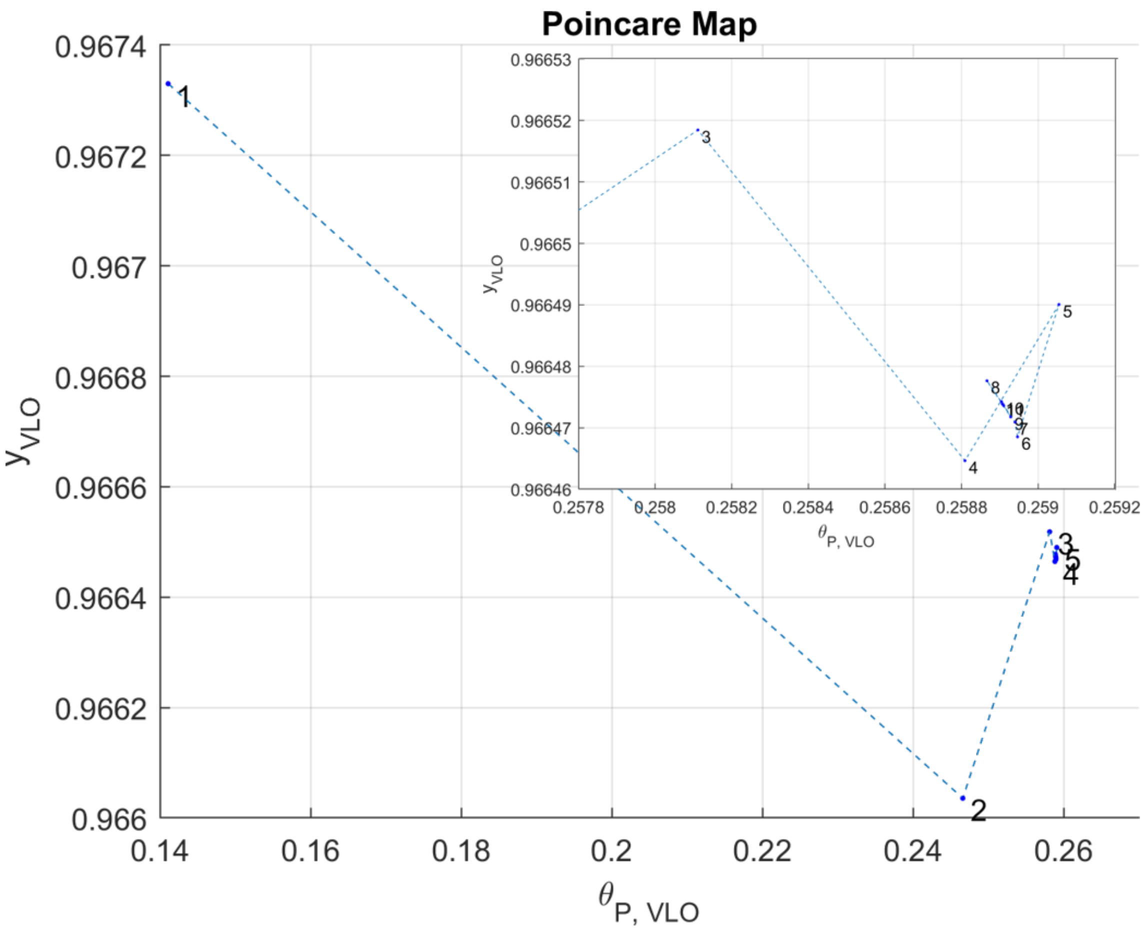


Figure 3.9: 2D section of the Poincaré Map where the numbers indicate the step number (zoomed in version is shown in the right upper corner of the figure)

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. This stiffness limits was chosen to have a similar height trends with [14] and [9] 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 [14] 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.

3.6 Conclusion

In this chapter, a template model called SLIP-SL was proposed which is an extension to the popular SLIP model. SLIP model can generate the reference CoM trajectories that can mimic the two-phased walking but it doesn't have swing leg dynamics. Thus, when this template model is used, additional steps are necessary for obtaining the swing foot trajectory so that the actual robot can be controlled.

Unlike the conventional method using SLIP model, the method using SLIP-SL model can also calculate the natural dynamics of swing leg so that it doesn't need to consider the additional step for obtaining the swing foot trajectory. From this study, we could get the result that using a simple model that is modified to suit the bipedal robots could improve the energy efficiency and reduce the effort for additional steps of obtaining the swing foot trajectory.

Chapter 4

Robust analysis of biped walking using output zeroing control with trajectory optimization

4.1 Introduction

In the research of the bipedal robot, the robustness of the gait is also important issue in control schemes. Since the environment that implements bipedal robot walking is not always ideal, the effects of steps and uneven terrain should be considered in the periodic walking.

In the previous research of the bipedal robots, the same issue occurred in a limit cycle walker's gait. This walker is the robot whose control methods is using the passive dynamics of the robot to improve the efficiency of energy consumption of the gait [16], [17]. These methods were motivated by the natural movement of the robot such as walking through the slope without using any of actuation power and realizing the walking only with the mechanical elements like spring or damper [18], and there were various studies for the efficient walkers based on this motivations [19] [20] [21]. However, this limit cycle walkers are susceptible to disturbances such as effect from uneven terrain or external forces. This limit cycle can be easily collapsed by small amount of these disturbances, so in the previous research, the control method with using output zeroing control to increase the robustness of the limit cycle walking was proposed [2] [22] [23]. This method uses the output function with a relative degree 3, which consists of the angular momentum and the function which is artificially constructed by integrating the state of the dynamics. Using this output function, the zero dynamics of the systems can be reduced by 1, so that it becomes easier to satisfy the condition of the stability of the whole system compared with using conventional output function with relative degree 2. This method was successfully implemented for walking of a 4-link biped robot and has been shown that the robustness of the limit cycle walking was increased by implementation with numerical simulation.

Since the periodic gait obtained by using the trajectory optimization method is easier to calculate and the application is simpler than the limit cycle method, we expanded this method to the gait of 4-link robot which is generated by a trajectory optimization. In this thesis, we represent how to design the output zeroing controller to reduce the dimension of zero dynamics, the method of optimizing trajectory, and trajectory interpolation to make the continuous output function. Then, we verified the robustness of the gait with numerical simulation compared with the previous research.

4.2 Dynamics model of 4-link biped robot

In this section, we describe the mathematical model of the dynamics of 4-link biped robot which is shown in Figure 4.1. This model consists of four links of rigid bodies with four joints. These links correspond to the upper leg and lower leg of the walker and each of joints has an actuator. We assumed that the tip of the stance leg is constrained to the ground, so that we can express this constraint as $\dot{x}_1 = 0$, and $\dot{z}_1 = 0$. Then, the dynamic equation of this 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 $\boldsymbol{q} = [\theta_1, \theta_2, \theta_3, \theta_4]^T$ is a generalized coordinate vector, $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{4\times4}$ is an inertia matrix, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{4\times4}$ is a Coriolis and centrifugal force matrix, $\boldsymbol{G}(\boldsymbol{q}) \in \mathbb{R}^4$ is a gravitational matrix, $\boldsymbol{u} \in \mathbb{R}^3$ is a input torque from actuators, and $\boldsymbol{S} \in \mathbb{R}^{4\times3}$ is a transformation matrix which can be written as

$$\boldsymbol{S} = \begin{bmatrix} -1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.2}$$

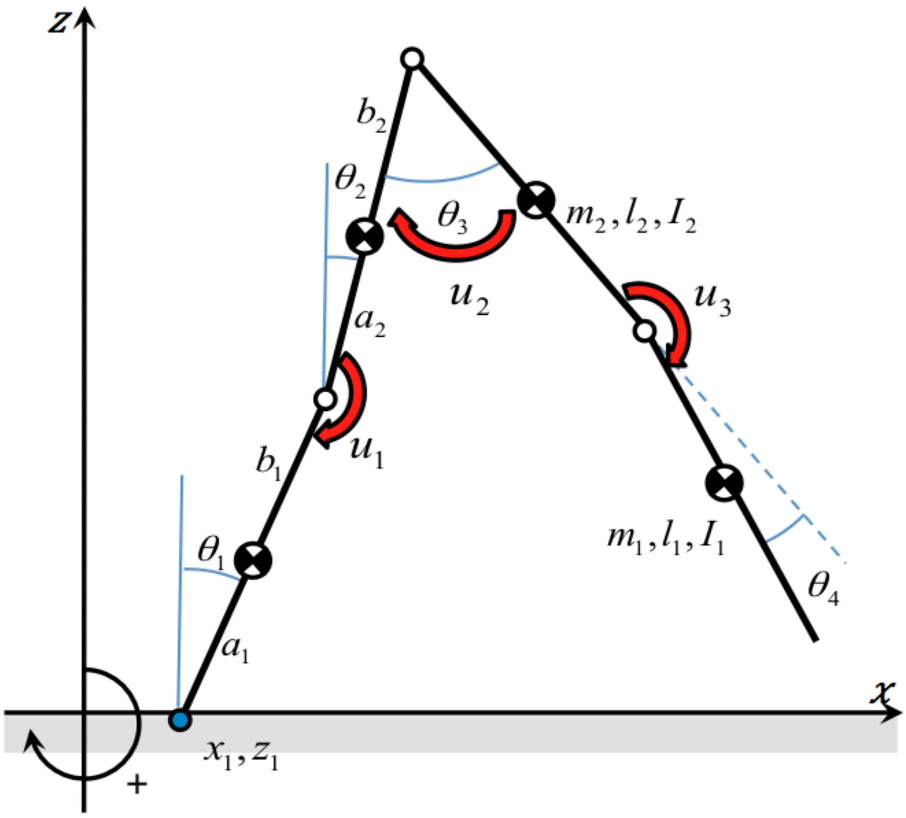


Figure 4.1: Model of 4link biped robot

4.3 Control law

4.3.1 Walking with holonomic constraints

In order to apply the output zeroing control with relative degree 3 to 4-link model, we apply holonomic constraints on both knee joints θ_2 and θ_4, given by

$$\begin{aligned}
\theta_2 &= \theta_1 + f_{v1}(\theta_3), && (4.3)\\
\theta_4 &= f_{v2}(\theta_3) && (4.4)\\
\dot{\theta}_2 &= \dot{\theta}_1 + \frac{\partial f_{v1}(\theta_3)}{\partial \theta_3}\dot{\theta}_3 && (4.5)\\
\dot{\theta}_4 &= \frac{\partial f_{v2}(\theta_3)}{\partial \theta_3}\dot{\theta}_3, && (4.6)
\end{aligned}$$

where $f_{v1}(\theta_3)$ and $f_{v2}(\theta_3)$ represent the holonomic constraints of the stance leg knee joint and the swing leg knee joint, respectively. Note that instead of imposing holonomic constraint on the absolute angle θ_2 of the stance leg knee joint, we impose a holonomic constraint on relative angle $\theta_2 - \theta_1$ of the supporting leg knee joint. Considering the constraint equations (4.3)˜(4.6), the dynamic equation can be reduced as

$$\boldsymbol{M}_L(\boldsymbol{q}_L)\ddot{\boldsymbol{q}}_L + \boldsymbol{C}_L(\boldsymbol{q}_L, \dot{\boldsymbol{q}}_L)\dot{\boldsymbol{q}}_L + \boldsymbol{G}_L(\boldsymbol{q}_L) = \boldsymbol{S_L u_2}, \qquad (4.7)$$

where $\boldsymbol{q}_L = [\theta_1, \theta_3]^T$, $\boldsymbol{S}_L \in \mathbb{R}^{2\times 1}$ is a transformation matrix.

4.3.2 Definition of angular momentum

The angular momentum of this system L at the contact point is given by

$$L = \boldsymbol{M}_L^{11}\dot{\boldsymbol{\theta}}_1 + \boldsymbol{M}_L^{12}\dot{\boldsymbol{\theta}}_3 := H(\theta_3)\dot{\boldsymbol{\theta}}_1 + I(\theta_3)\dot{\boldsymbol{\theta}}_3, \qquad (4.8)$$

where $\boldsymbol{M}_L^{ij}$ is a (i,j) element of $\boldsymbol{M}_L(\boldsymbol{q}_L)$. A time derivative of the angular momentum L can be obtained as the following by taking the product of the walking direction component of the center of mass of the entire system and total weight of the system as

$$\begin{aligned} \dot{L} &= e_1\sin(\theta_1) + e_2\sin(\theta_1 + f_{v1}) + e_3\sin(\theta_1 + f_{v1} - \theta_3) \\ &\quad + e_4\sin(\theta_1 + f_{v1} - \theta_3 + f_{v2}), \end{aligned} \tag{4.9}$$

where e_1 ˜e_4 are the constants determined by the mechanical parameters of this model. With the angular momentum L, a new function p is defined by

$$p = \theta_1 + \omega(\theta_3), \tag{4.10}$$

$$\omega(\theta_3) = \int_{\theta_3(0)}^{\theta_3(t)} \frac{I(\theta_3)}{H(\theta_3)} d\theta_3. \tag{4.11}$$

4.3.3 Output zeroing control with relative degree 3

When a state variable is selected as $x = [p, L, \theta_3, \dot{\theta}_3]^T$, a state equation of this system can be written as

$$\begin{bmatrix} \dot{p} \\ \dot{L} \\ \dot{\theta}_3 \\ \ddot{\theta}_3 \end{bmatrix} = \begin{bmatrix} L/H(\theta_3) \\ G_p(p,\theta_3) \\ \dot{\theta}_3 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u_b, \tag{4.12}$$

where,

$$\begin{aligned} G_p(p,\theta_3) &:= e_1\sin(p - \omega(\theta_3)) + e_2\sin(p - \omega(\theta_3) + f_{v1}) \\ &\quad + e_3\sin(p - \omega(\theta_3) + f_{v1} - \theta_3) \\ &\quad + e_4\sin(p - \omega(\theta_3) + f_{v1} - \theta_3 + f_{v2}), \end{aligned}$$

and $u_b := \ddot{\theta}_3$ is a new input of this system.
We define an output function y as

$$y = L + \alpha_1 p, \tag{4.13}$$

where α_1 is a weight parameter. In this output function, input-output relative degree becomes 3 for the system due to the time derivative of y is given as

$$\begin{aligned}
y^{(3)} &= L^{(3)} + \alpha_1 p^{(3)}, && (4.14)\\
\ddot{L} &= \frac{dG_p}{dt} = \frac{\partial G_p}{\partial p}\frac{L}{H} + \frac{\partial G_p}{\partial \theta_3}\dot{\theta}_3, && (4.15)\\
L^{(3)} &= \frac{d}{dt}\left(\frac{\partial G_p}{\partial p}\frac{L}{H}\right) + \frac{d}{dt}\left(\frac{\partial G_p}{\partial \theta_3}\right)\dot{\theta}_3 + \frac{\partial G_p}{\partial \theta_3}u_b, && (4.16)\\
\ddot{p} &= \frac{G_p}{H} - \frac{(\partial H/\partial \theta_3)L}{H^2}\dot{\theta}_3, && (4.17)\\
p^{(3)} &= \frac{d}{dt}\left(\frac{G_p}{H}\right) - \frac{d}{dt}\left(\frac{(\partial H/\partial \theta_3)L}{H^2}\right)\dot{\theta}_3 - \frac{(\partial H/\partial \theta_3)L}{H^2}u_b. && (4.18)
\end{aligned}$$

From the above, we can determine $y^{(3)}$ by controlling u_b. Therefore, for the reference output function y_d, we define $e := y - y_d$ and a new control input is given by

$$v = -\alpha_2 \ddot{e} - \alpha_3 \dot{e} - \alpha_4 e + y_d^{(3)}, \tag{4.19}$$

where α_2 ˜α_4 are gain parameters. Other robust controls also can be applied. Equation (4.19) can be transformed into

$$e^{(3)} + \alpha_2 \ddot{e} + \alpha_3 \dot{e} + \alpha_4 e = 0. \tag{4.20}$$

If α_2 ˜α_4 are determined as equation (4.20) is stable, e converges to 0 exponentially, i.e., the output y is tracked on the desired trajectory

y_d. Control intput u_b is obtained as

$$\begin{aligned} u_b &= \left(v - \frac{d}{dt}\left(\frac{\partial G_p}{\partial p}\frac{L}{H}\right) - \frac{d}{dt}\left(\frac{\partial G_p}{\partial \theta_3}\right)\dot{\theta}_3 - \alpha_1\frac{d}{dt}\left(\frac{G_p}{H}\right)\right. \\ &\quad \left. +\alpha_1\frac{d}{dt}\left(\frac{(\partial H/\partial \theta_3)L}{H^2}\right)\dot{\theta}_3\right)\left(\frac{\partial G_p}{\partial \theta_3} - \alpha_1\frac{(\partial H/\partial \theta_3)L}{H^2}\right)^{-1} \end{aligned} \tag{4.21}$$

In order to transform u_b into u, we use the following relationship:

$$\begin{aligned} u_b &= \ddot{\theta}_3 \\ &= \boldsymbol{I}_r\ddot{\boldsymbol{q}} \\ &= \boldsymbol{I}_r\boldsymbol{M}(\boldsymbol{q})^{-1}(\boldsymbol{S}\boldsymbol{u} - \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{G}(\boldsymbol{q})), \end{aligned} \tag{4.22}$$

where $\boldsymbol{I}_r = [0, 0, 1, 0]$.

4.3.4 Control for realizing holonomic constraints

In order to use the control law in the previous section, the trajectories of θ_2 and θ_4 in walking need to satisfy holonomic constraints equation (4.3), (4.4). Therefore, we use output zeroing control with relative degree 2 to control θ_2 and θ_4.

When a state variable is selected by $\boldsymbol{\xi} = \left[\boldsymbol{q}^T, \dot{\boldsymbol{q}}^T\right]^T$, a state equation of this system is given by

$$\begin{aligned} \dot{\boldsymbol{\xi}} &= \begin{bmatrix} \dot{\boldsymbol{q}} \\ -\boldsymbol{M}(\boldsymbol{q})(\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q})) \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{4\times 3} \\ \boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{S} \end{bmatrix}\boldsymbol{u} \\ &= f(\boldsymbol{\xi}) + g(\boldsymbol{\xi})\boldsymbol{u}. \end{aligned} \tag{4.23}$$

We define the output function h for the system as

$$\boldsymbol{h} = \begin{bmatrix} \theta_1 - \theta_2 \\ \theta_4 \end{bmatrix}. \tag{4.24}$$

Input-output relative degree becomes 2 for the system due to the time derivative of $\boldsymbol{h}$. Therefore, for the holonomic constraints $\boldsymbol{h}_d$, we define $\epsilon := \boldsymbol{h} - \boldsymbol{h}_d$ and a new control input is determined as

$$\boldsymbol{\nu} = -\beta_1\dot{\boldsymbol{\epsilon}} - \beta_2\boldsymbol{\epsilon} + \ddot{\boldsymbol{h}}_d, \tag{4.25}$$

where β_1 and β_2 are constants. Other robust controls also can be applied. Equation (4.25) is transformed into

$$\ddot{\boldsymbol{\epsilon}} + \beta_1\dot{\boldsymbol{\epsilon}} + \beta_2\boldsymbol{\epsilon} = 0. \tag{4.26}$$

If β_1 and β_2 are determined so that equation (4.26) is stable, $\boldsymbol{\epsilon}$ converges to 0 exponentially, the output $\boldsymbol{h}$ tracks the desired trajectory $\boldsymbol{h}_d$. The control input $\boldsymbol{u}_b$ is obtained as

$$\boldsymbol{\nu} = L_f^2\boldsymbol{h} + L_gL_f\boldsymbol{h}, \tag{4.27}$$

where L_f and L_g are Lie derivative of f and g, respectively. From the equation (4.22) and (4.27), the actual control input $\boldsymbol{u}$ can be determined as follows:

$$\begin{aligned} u \; = \; & \begin{bmatrix} L_gL_f\boldsymbol{h} \\ \boldsymbol{I}_r\boldsymbol{M}\left(\boldsymbol{q}\right)^{-1}\boldsymbol{S} \end{bmatrix}^{-1} \\ & \begin{bmatrix} \boldsymbol{\nu} - L_f^2\boldsymbol{h} \\ \boldsymbol{u}_b + \boldsymbol{I}_r\boldsymbol{M}\left(\boldsymbol{q}\right)^{-1}\left(\boldsymbol{C}\left(\boldsymbol{q},\dot{\boldsymbol{q}}\right)\dot{\boldsymbol{q}} + \boldsymbol{G}\left(\boldsymbol{q}\right)\right) \end{bmatrix}. \end{aligned} \tag{4.28}$$

4.4 Direct collocation method

In order to apply the control law as shown in the previous section, we need to obtain the reference output function y_d and the holonomic constraints h_d.In this chapter, we solve the following trajectory optimization problem to generate the walking gait,

$$\begin{aligned} \min_{Z} \ & \bar{J} \\ s.t. \ & \dot{x} = \bar{f}(x,u) \\ & \bar{P}(x,u) \leq 0 \\ & \bar{B}(x,u) \leq 0 \\ & T(x,c) \leq 0 \end{aligned} \tag{4.29}$$

where $x \in X, u \in U, z := [x^T, u^T, c^T]^T \in Z$. Note that $x \in X$, $u \in U$, c are the states, inputs and parameters of output function at the walking gait, respectively, and $z \in Z$ is optimization variables. Note that $\bar{J}$ is a cost function, $\bar{f}$ represents dynamics of the robot, $\bar{P}(x,u)$, $\bar{B}(x,u)$ and $T(x,c)$ are constraints that give path constraints, boundary constraints, and reference output and holonomic constraints, respectively. To solve the above problem, we used Direct collocation method [12] which discretizes a continuous time system and solves the continuous time optimization problem of equation (4.29) as a nonlinear programming problem as

$$\begin{aligned} \min_{Z} \ & J \\ s.t. \ & x_{k+1} = f(x_k, u_k) \\ & P(x_k, u_k) \leq 0 \\ & B(x_k, u_k) \leq 0 \\ & T(x_k, c) \leq 0 \end{aligned} \tag{4.30}$$

where,$X := [x_0, ..., x_F]^T, U := [u_0, ..., u_F]^T, z_k := [x_k, u_k, c^T]^T$,and $Z := [z_0, ..., z_F]^T$. And, we used the Hermite-Simpson collocation method to interpolate the discrete points by Hermite splines with using approximation of the integral calculation by the Simpson's rule. Hermite-Simpson collocation method has good solution accuracy and approximates the trajectory of the state with 3 order Hermite spline so that the time derivative is continuous.

4.4.1 Dynamics for discretized system

Direct collocation method defines a constraint equation to approximate the dynamics of the system. In the Hermite-Simpson collocation method, we rewrite the dynamics to equation (4.33) by using Simpson's rule:

$$\begin{aligned} \dot{x} &= f(t) && (4.31) \\ \int_{t_k}^{t_{k+1}} \dot{x} dt &= \int_{t_k}^{t_{k+1}} f(t)\, dt && (4.32) \\ x_{k+1} - x_k &= \frac{1}{6} h_k \left(f_k(t) + 4 f_{k+\frac{1}{2}}(t) + f_{k+1}(t) \right), && (4.33) \end{aligned}$$

where $h_k = t_{k+1} - t_k$. Dynamics of the midpoint between the discrete points is required, such as $f_{k+\frac{1}{2}}$ in the equation (4.33). Therefore, by preparing the interpolation function of the state trajectory [12], the intermediate point is calculated as

$$x_{k+\frac{1}{2}} = \frac{1}{2}(x_k + x_{k+1}) + \frac{h_k}{8}(f_k - f_{k+1}). \quad (4.34)$$

4.4.2 Path constraints

Path constraints are constraints at all discrete points. In this thesis, following 2 constraints are given:

- Swing leg position constraint
  Constraint that prevents swing legs from colliding with the ground except at the terminal point:

$$p^z_{sw} > 0. \tag{4.35}$$

- Constraint of constraint force
  Constraint that prevents lifting of the contact foot from the ground at the termination point. In this case, we assume that the robot does not slip from the contact point. Therefore, the constraint of the constraint force is as:

$$\lambda_z > 0, \tag{4.36}$$

  where λ_z is the vertical component of the constraint force.

4.4.3 Boundary constraints

Boundary constraints are constraints at the initial point and the terminal point. In this chapter, following 4 constraints are given:

- Stance leg position constraint
  Constraint for the stance leg to contact the ground during walking and not move form the contact point:

$$x_{1,0} = z_{1,0} = 0, \tag{4.37}$$

$$\dot{x}_{1,0} = \dot{z}_{1,0} = 0. \tag{4.38}$$

- Step length constraint
  Constraint for making a step length of the robot at an arbitrary value l:

$$p^x_{sw,F} - x_{1,0} = l, \tag{4.39}$$

where $p^x_{sw,F}$ is the horizontal component of the swing leg tip position at the terminal point.

- Periodic orbital constraint
  Since the terminal point is in the state just before the collision, a reset map is applied to the terminal point to match it with the initial point as:
  $$x_0 = f_H\left(x_F\right), \tag{4.40}$$
  where f_H is the reset map.

- Constraint on instantaneous leg changing is given by:

$$\dot{p}^z_{sw,0} > 0, \tag{4.41}$$
$$\dot{p}^z_{sw,F} < 0. \tag{4.42}$$

4.4.4 Reference output and holonomic constraints

Since the output function in the walking gait derived by the direct collocation method is defined by the discrete time, it can not be used as the reference output function as it is. In this thesis, an approximate output function is defined as Bézier curve given by

$$b\left(s, c\right) := \sum_{k=0}^{M} c_k \frac{M!}{k!\left(M-k\right)!} s^k \left(1-s\right)^{M-k}. \tag{4.43}$$

Since the output function needs to have 3rd order time derivative when applying output zeroing control with relative degree 3, the reference output function approximated by Bézier curve requires accuracy up to 3rd order time derivative. Also, we use p_k defined by the equation (4.10) as an argument of Bézier curve. However, since the domain of

Bézier curve is $[0, 1]$, we actually use the variable $s_{1,k}$ normalized to p_k as:

$$s_{1,k} = \frac{p_k - p_0}{p_F - p_0}. \tag{4.44}$$

Therefore, constraints deriving the reference output function are summarized by

$$\begin{aligned} b\left(s_{1,0}, c\right) &= y_0 && (4.45)\\ \dot{b}\left(s_{1,0}, \dot{s}_{1,0}, c\right) &= \dot{y}_0 && (4.46)\\ \ddot{b}\left(s_{1,0}, \dot{s}_{1,0}, \ddot{s}_{1,0}, c\right) &= \ddot{y}_0 && (4.47)\\ b^{(3)}\left(s_{1,k}, \dot{s}_{1,k}, \ddot{s}_{1,k}, s_{1,k}^{(3)}, c\right) &= y_k^{(3)}, && (4.48) \end{aligned}$$

and p_k needs to be monotonically increasing:

$$\dot{p}_k > 0. \tag{4.49}$$

Similar to deriving the reference output function, holonomic constraints f_{v1} and f_{v2} are also derived as Bézier curves. The argument of these holonomic constraints is θ_3 and we actually use the variable $s_{1,k}$ normalized to p_k as:

$$s_{2,k} = s_{3,k} = \frac{\theta_{3,k} - \theta_{3,0}}{\theta_{3,F} - \theta_{3,0}}. \tag{4.50}$$

Therefore, constraints deriving the holonomic constraints are summarized by

$$\begin{aligned} f_{v1}\left(s_{2,0}, c\right) &= \theta_{2,0} - \theta_{1,0} && (4.51)\\ \dot{f}_{v1}\left(s_{2,0}, \dot{s}_{2,0}, c\right) &= \dot{\theta}_{2,0} - \dot{\theta}_{1,0} && (4.52)\\ \ddot{f}_{v1}\left(s_{2,k}, \dot{s}_{2,k}, \ddot{s}_{2,k}, c\right) &= \ddot{\theta}_{2,k} - \ddot{\theta}_{1,k} && (4.53) \end{aligned}$$

$$\begin{aligned} f_{v2}\left(s_{3,0}, c\right) &= \theta_{4,0} & (4.54)\\ \dot{f}_{v2}\left(s_{3,0}, \dot{s}_{3,0}, c\right) &= \dot{\theta}_{4,0} & (4.55)\\ \ddot{f}_{v2}\left(s_{3,k}, \dot{s}_{3,k}, \ddot{s}_{3,k}, c\right) &= \ddot{\theta}_{4,k}, & (4.56)\end{aligned}$$

and θ_3 needs to be monotonically increasing:

$$\dot{\theta}_{3,k} > 0. \tag{4.57}$$

4.5 Simulation results

In this section, we present the numerical simulation results using the above method. The walking gait converged to a stable limit cycle as shown in Figure 4.2˜4.5. These diagram show the phase of the state where the 4-link robot walks for 12 steps after they converged to the limit cycle.

4.5.1 Disturbance rejection performance

In order to verify the effectiveness of the proposed method, we compared the robustness with a previous method [2]. We used a Gait Sensitivity Norm (GSN) [24] and H_∞ norm [2] as a comparison factors. In this study, the gait period T is used as the gait indicator like [24]. The state perturbation δ is set to 0.1 % of the value at the equilibrium point $\bar{v}$ of each state. Table. 4.1 shows the GSN and H_∞ norm, the robustness is better when they are smaller, of the previous method and the proposed method. We could get the result that the proposed method using the reference output function which is calculated by the direct collocation method improves the robustness quantitatively.

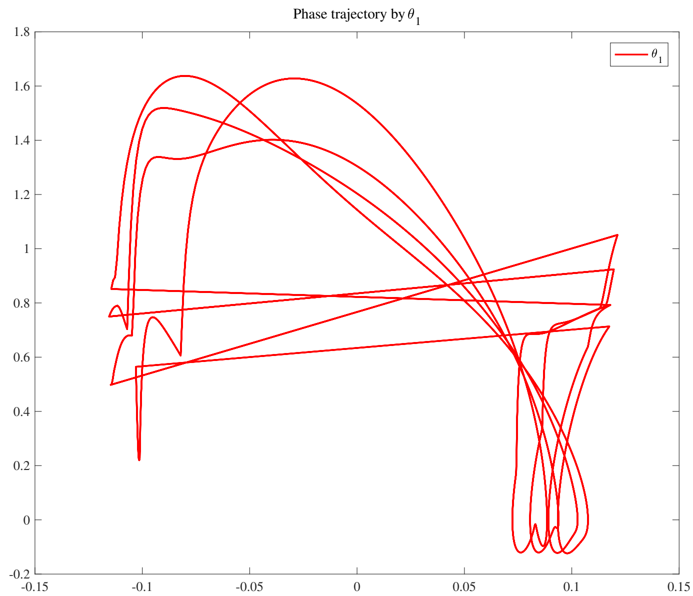


Figure 4.2: Phase trajectory of θ_1 in the walking simulation

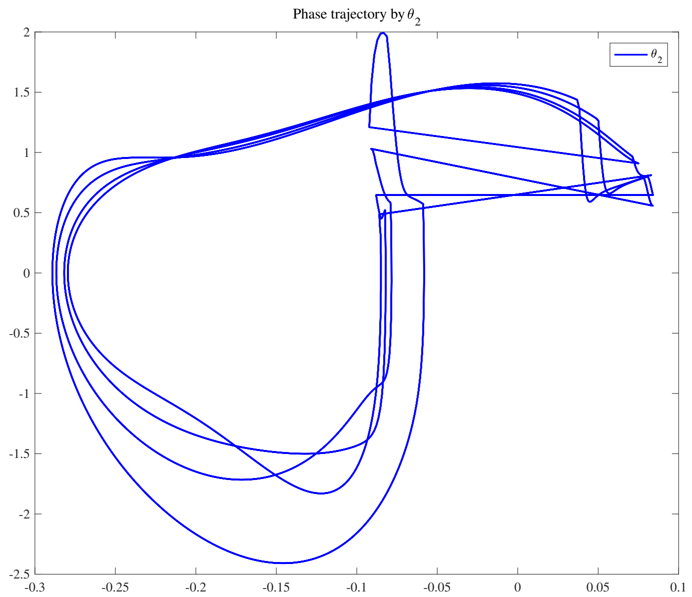
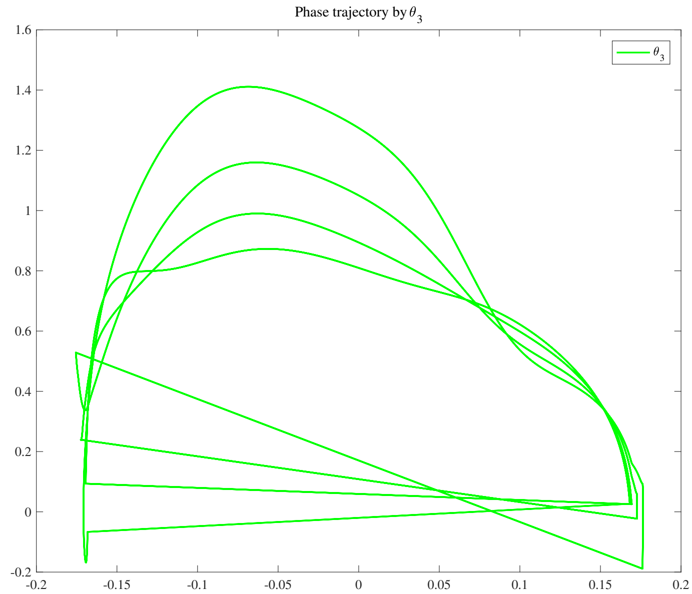


Figure 4.3: Phase trajectory of θ_2 in the walking simulation



.

Figure 4.4: Phase trajectory of θ_3 in the walking simulation

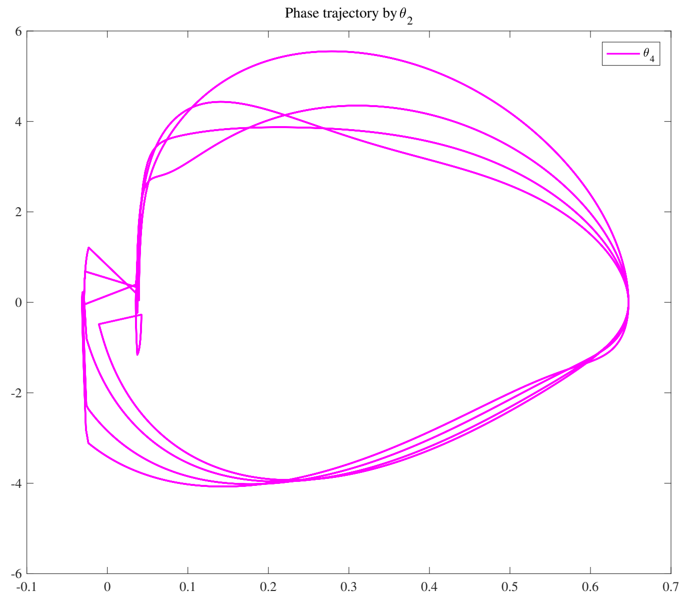


Figure 4.5: Phase trajectory of θ_4 in the walking simulation

Table 4.1: GSN and H_∞ norm

	GSN	H_∞ norm
Previous Method	21.9	22.9
Proposed Method	13.2	13.6

4.5.2 Zero dynamics trajectory

From the above result, we could get the fact that the proposed method improves the robustness of the 4-link robot's walk. One of the reasons why the proposed method improves the robustness is that the zero dynamics follows the desired trajectory. Figure 4.6 and Figure 4.7 show the error between the zero dynamics p and the desired trajectory $\bar{p}$ in the limit cycle in the previous method and the proposed method, respectively.

4.6 Conclusion

In this chapter, we proposed a method to derive a reference output function to be used in output zeroing control method with a relative degree 3 by direct collocation method. We conducted a numerical simulation and verified that the proposed method increases the robustness on uneven terrain compared with the previous method. When the output function is designed with a relative degree 3, this reduces the dimension of zero dynamics of the system as 1. Consequently, using this virtual simple template model resulted in the improvement of the robustness of bipedal walking by modifying the model to reduce the dimension of zero dynamics, and we verified that this result can be expanded to the gait which is obtained by the trajectory optimization.

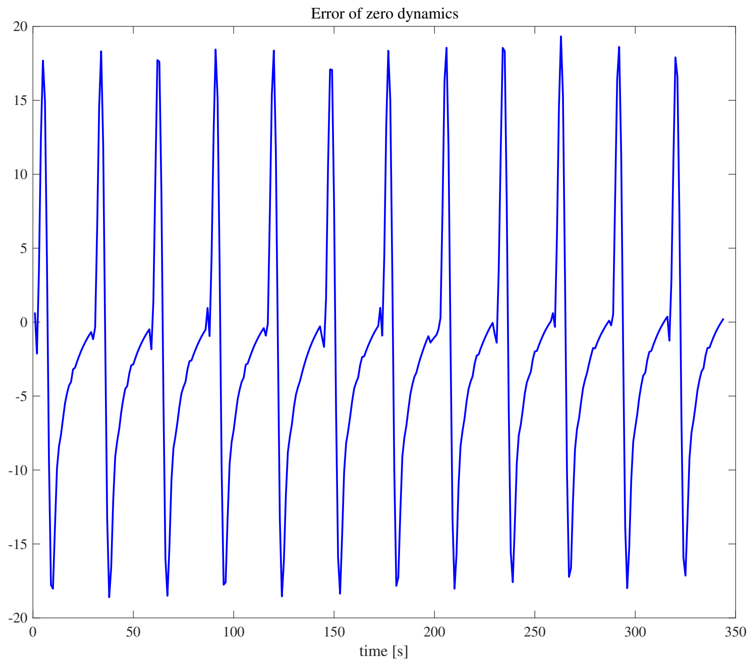


Figure 4.6: Error of zero dynamics trajectory in the previous method

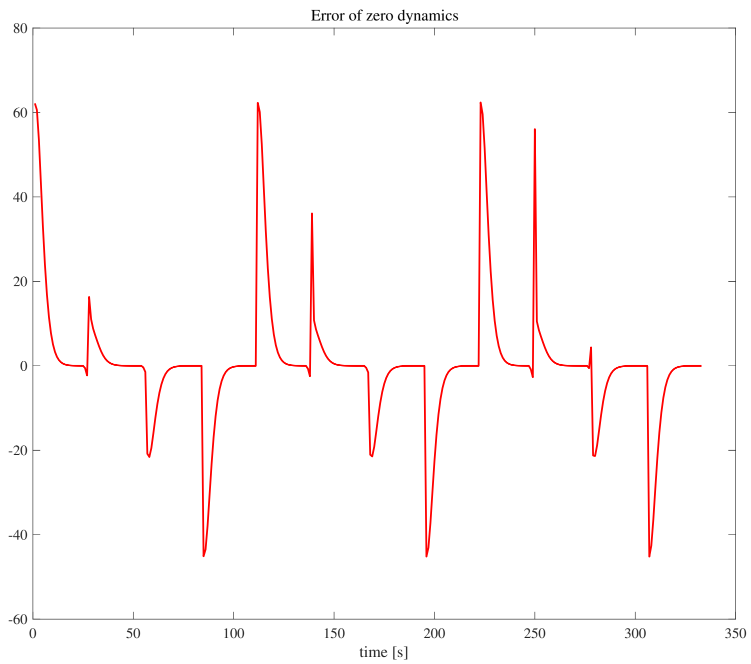


Figure 4.7: Error of zero dynamics trajectory in the proposed method

Chapter 5

Running control of bipedal robot with series elastic actuator

5.1 Introduction

In recent years, various studies has been developed in the field of bipedal robots for both the control methods and the mechanical improvement of the robots. About the control methods for the bipedal robot, some of them are inspired by a natural simple dynamics such as an inverted pendulum model [25], a spring-loaded inverted pendulum (SLIP) model [26] [27] as described in chapter 2 and 3, and others use the natural motions of creatures such as a human's walking patterns [28] and so on. These ideas are based on the goals to overcome the difficulties in common use like the energy efficiency of the motors or the durability of the robot's body against high frequency impact forces. As the efforts for these goals, many improvements in the structural point of view also have been proposed.

In particular, the robots with the series elastic actuator (SEA) have been focused in recent researches, whose mechanism is composed of a motor and a link and there exist elasticity elements between them. This mechanism was adopted for the reasons that it has the adaptability in the unknown environment [3], it can storage the energy for the robot's hopping [4], and it also reduce the possibility of the damages in motors which are caused by high frequency impact forces by absorbing the impact forces by elastic elements. From these advantages, the SEA robots are expected to realize high-energy efficiency running gaits and the high-durability from the impact forces.

The dynamics of a robot with this mechanical structure is usually more complex than robots without this structure, a conventional control method for non-SEA robot generally could not cover for realizing the movement of SEA robots. As a well-known control method for a SEA robot is the output zeroing control which is based on using the joint angle as an output function and omitting the inertial coupling terms between the link connected to the motor and the joint link of the robot, so that the relative degree of this system is assumed to be four. [29] and [30] uses this method to control the robot whose joints are compliantly actuated, and the result of this method was successfully implemented for their target systems. However, this method can be used only when the input term does not appear until the output function is differentiated four times, so it is difficult to apply it to model with a structure which the inertial coupling term cannot be ignored.

In this chapter, we present the control method to realize the running of a bipedal robot which consists of a series elastic structure. The target system is assumed to have the inertial coupling term between the link of a motor and the link of the joint, so adopting the conventional control method is quite difficult. To overcome this problem, we use a model matching control method with the running gait of a non-SEA robot model which uses the SLIP model as a template model.

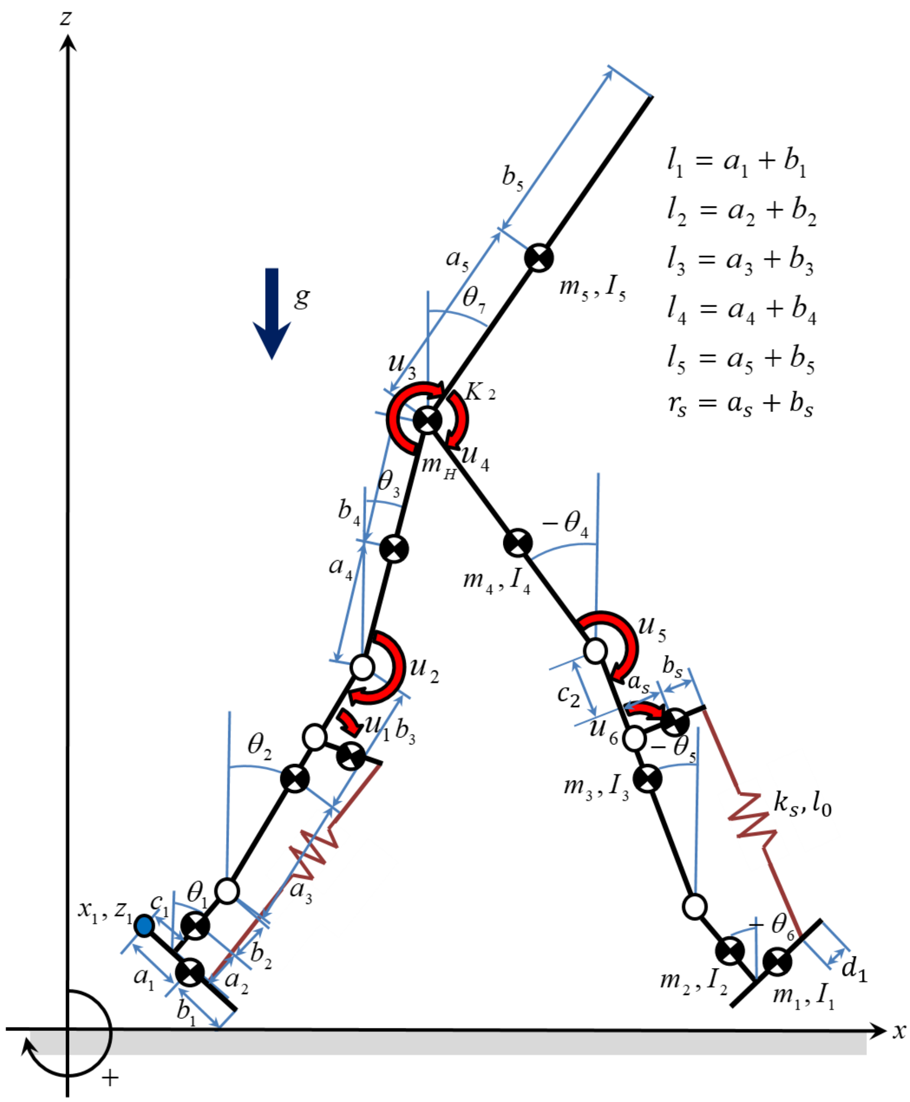


Figure 5.1: Planar bipedal robot model

5.2 Dynamic model of running robot

In this section, we describe the dynamic model of running robot with series elastic actuator.

5.2.1 Kinetic and potential energy

In this study, the target model consists of upper body, two upper legs, two lower legs, two flat feet and two additional links which is fixed on lower legs with motors to transfer the torques to control the ankle joints, and these additional links are combined to the feet links with linear spring, respectively. Figure 5.1 shows the conceptual diagram of the target system. Unlike conventional research of SEA mechanisms, we also consider the reaction force between the motor's link and lower leg's link. This condition allow us to get the kinetic energy of rigid body model as

$$T = \frac{1}{2}\begin{bmatrix} \dot{\boldsymbol{q}}^T & \dot{\boldsymbol{\phi}}^T \end{bmatrix}\begin{bmatrix} \boldsymbol{M}(\boldsymbol{q},\boldsymbol{\phi}) & \boldsymbol{A}(\boldsymbol{q},\boldsymbol{\phi}) \\ \boldsymbol{A}(\boldsymbol{q},\boldsymbol{\phi})^T & \boldsymbol{B} \end{bmatrix}\begin{bmatrix} \dot{\boldsymbol{q}} \\ \dot{\boldsymbol{\phi}} \end{bmatrix}, \tag{5.1}$$

where $\boldsymbol{q} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7, x_1, z_1]^T$ is a generalized coordinate vector, which represent the joint angles and the position of the heel in the support leg, $\boldsymbol{\phi} = [\phi_1, \phi_2]^T$ are the motor angles which drive SEA elements. $\boldsymbol{M}(\boldsymbol{q},\boldsymbol{\phi}) \in \mathbb{R}^{9\times9}$ is the link inertia matrix, $\boldsymbol{B} \in \mathbb{R}^{2\times2}$ is the motor inertia diagonal matrix, and $\boldsymbol{A}(\boldsymbol{q},\boldsymbol{\phi}) \in \mathbb{R}^{9\times2}$ is the inertial coupling matrix between the motor and the lower leg link of the SEA. And the potential energy for this system can be written as

$$U = U_g(\boldsymbol{q},\boldsymbol{\phi}) + U_s(\boldsymbol{q},\boldsymbol{\phi}), \tag{5.2}$$

where $U_g(\boldsymbol{q},\boldsymbol{\phi})$ is the gravitational energy and $U_s(\boldsymbol{q},\boldsymbol{\phi})$ is the elastic energy of the spring in the SEA mechanism. In particular, $U_s(\boldsymbol{q},\boldsymbol{\phi})$ is calculated as

$$U_s(\boldsymbol{q},\boldsymbol{\phi}) = \sum_{i=1}^{2}\frac{1}{2}k_{s,i}(l_{s,i} - l_0)^2, \tag{5.3}$$

where $k_{s,i}, i \in 1,2$ is the spring constant of the SEA, l_0 is natural length of the spring, and $l_{s,i}$ is the length of the spring which is calculated by a distance between the link's end of the motor which drives the SEA and the position of the spring of the SEA fixed on the foot.

5.2.2 Dynamics of running robot

Using the derivation of Eular-Lagrange equation with constraint conditions, the dynamics of above model can be written as

$$\boldsymbol{M}_a(\boldsymbol{q}_a)\ddot{\boldsymbol{q}}_a + \boldsymbol{C}(\boldsymbol{q}_a, \dot{\boldsymbol{q}}_a) = \boldsymbol{S}\boldsymbol{u} + \boldsymbol{J}_c(\boldsymbol{q})^T\boldsymbol{\lambda}, \tag{5.4}$$

where $\boldsymbol{q}_a := [\boldsymbol{q}^T, \boldsymbol{\phi}^T]^T$ is a generalized coordinate vector which represents whole state elements of the system.$\boldsymbol{C}(\boldsymbol{q}_a, \dot{\boldsymbol{q}}_a) \in \mathbb{R}^{11\times 1}$ contains centrifugal, Coriolis, gravity and spring forces, $\boldsymbol{S} \in \mathbb{R}^{11\times 6}$ is a transformation matrix between motor inputs and generalized forces, $\boldsymbol{J}_c(\boldsymbol{q}) \in \mathbb{R}^{P\times 11}$ is a Jacobian matrix of the constraint, P is a number of constraints, and $\boldsymbol{\lambda} \in \mathbb{R}^P$ is a constraint force vector given by

$$\begin{aligned}
\boldsymbol{\lambda} &= -\boldsymbol{X}(\boldsymbol{q}_a)^{-1}(\boldsymbol{J}_c(\boldsymbol{q})\boldsymbol{M}_a(\boldsymbol{q}_a)^{-1}\boldsymbol{\Gamma}(\boldsymbol{q}_a, \dot{\boldsymbol{q}}_a) + \dot{\boldsymbol{J}}_c(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}_a), && (5.5)\\
\boldsymbol{X}(\boldsymbol{q}_a) &= \boldsymbol{J}_c(\boldsymbol{q})\boldsymbol{M}_a(\boldsymbol{q}_a)^{-1}\boldsymbol{J}_c(\boldsymbol{q})^T, && (5.6)\\
\boldsymbol{\Gamma}(\boldsymbol{q}_a, \dot{\boldsymbol{q}}_a) &= \boldsymbol{S}\boldsymbol{u} - \boldsymbol{C}(\boldsymbol{q}_a, \dot{\boldsymbol{q}}_a). && (5.7)
\end{aligned}$$

Since we consider the running gait of the biped robot, there exist two continuous phases that called a stance phase and a flight phase, respectively. The stance phase is defined when the robot's support leg touches and stays on the ground while the flight phase is defined when the robot's two legs are both left from the ground. Therefore, the Jacobian matrix $\boldsymbol{J}_c(\boldsymbol{q})$ is different depending on each phase.

Stance phase.

Since the robot's feet is assumed to be flat, there exist three contact conditions as follows.

1) Heel contact : This contact condition is defined as the state when the heel of the support leg is staying on the ground and the following constraint conditions should be satisfied:

$$\dot{x}_1 = 0, \dot{z}_1 = 0. \tag{5.8}$$

2) Sole contact : This contact condition is defined as the state when both heel and toe of the support leg are staying on the ground. To constraint both heel's and toe's positions, the following constraint conditions should be satisfied:

$$\dot{x}_1 = 0, \dot{z}_1 = 0, -l_1 cos(\theta_1)\dot{\theta}_1 + \dot{z}_1 = 0. \tag{5.9}$$

3) Toe contact : This contact condition is defined as the state when the toe of the support leg is staying on the ground:

$$-l_1 sin(\theta_1)\dot{\theta}_1 + \dot{x}_1 = 0, -l_1 cos(\theta_1)\dot{\theta}_1 + \dot{z}_1 = 0. \tag{5.10}$$

Flight phase.

In the flight phase, there is no constraint force from the ground. Therefore, the second term on the right hand side of equation (5.4) can be assumed to be 0 ($\boldsymbol{J}_c(\boldsymbol{q})^T \boldsymbol{\lambda} = \boldsymbol{0}^{11\times1}$).

5.2.3 Modeling of state change at impact

When the support leg touches the ground, velocities of each joint are discontinuously changed. For simplicity, we assume that the collision between the support leg and the ground is completely inelastic so that the joint velocity after the collision $\dot{\boldsymbol{q}}_a^+$ can be considered as a mapping of the joint velocity before the collision $\dot{\boldsymbol{q}}_a^-$. In particular, the following impact equation is introduced:

$$\begin{aligned} \boldsymbol{\lambda}_I &= -\boldsymbol{X}_I(\boldsymbol{q}_a)^{-1}\boldsymbol{J}_I(\boldsymbol{q})\dot{\boldsymbol{q}}_a^-, & (5.11)\\ \boldsymbol{X}_I(\boldsymbol{q}_a) &= \boldsymbol{J}_I(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q}_a)^{-1}\boldsymbol{J}_I(\boldsymbol{q})^T, & (5.12)\\ \dot{\boldsymbol{q}}_a^+ &= (\boldsymbol{I}-\boldsymbol{M}(\boldsymbol{q}_a)^{-1}\boldsymbol{J}_I(\boldsymbol{q})^T\boldsymbol{X}_I(\boldsymbol{q}_a)^{-1}\boldsymbol{J}_I(\boldsymbol{q}))\dot{\boldsymbol{q}}_a^-. & (5.13) \end{aligned}$$

where $\boldsymbol{J}_I(\boldsymbol{q})$ is a Jacobian matrix derived from the constraint condition that was mentioned above. In this study, we assume that there is one SEA actuator in each leg, however, the following method can be easily extended to multiple SEA cases.

5.3 Control method

In the conventional researches [31], [32], control laws for stable running gaits were proposed. This method is based on a non-SEA robot model, which has nine rigid links without elasticity, six actuators at each joint. The study has shown that we can easily realize stable running gaits with desired steady forward velocities. Unfortunately, the SEA robot model which is studied in this chapter has some difficulties to realize the proposed this control law. In this section, we will show why these difficulties exist in SEA robot models and propose a method how to realize the stable running gaits of the non-SEA robot models.

5.3.1 Achieving the running gait

For realizing stable running gaits of non-SEA robot models, SLIP model based control is used frequently [26], [33], [31]. This control method defines reference accelerations at each joint in the stance phase so that each joint torque can be calculated to fit the net force on the robot's center of mass (CoM) to the reference acceleration. In this chapter, we refer the method as the sequential-contact bipedal running based on the SLIP model [31]. This work contains calculating the input to realize SLIP-like CoM movement and zero moment point (ZMP) control to handle the contact conditions in the stance phase using the input-output linearization method. Specifically, consider the following the dynamics of non-SEA robot model:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \bar{\boldsymbol{C}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \bar{\boldsymbol{S}}\bar{\boldsymbol{u}} + \boldsymbol{J}_c(\boldsymbol{q})^{\mathrm{T}}\bar{\boldsymbol{\lambda}}, \tag{5.14}$$

where $\bar{\boldsymbol{C}} \in \mathbb{R}^{9\times 1}$ contains centrifugal,Coriolis and gravity forces, $\bar{\boldsymbol{S}} \in \mathbb{R}^{9\times 6}$ is a transformation matrix between the motor inputs $\bar{\boldsymbol{u}} \in \mathbb{R}^{6\times 1}$ and the generalized forces. In this case, $\bar{\boldsymbol{u}}$ corresponds to torques of each hip,knee and ankle. $\bar{\boldsymbol{\lambda}} \in \mathbb{R}^P$ is a constraint force vector given as (5.5) to (5.7) with replacement of $\boldsymbol{q}_a$ to $\boldsymbol{q}$.

To calculate the input to realize SLIP-like CoM movement for the system (5.14), consider the position of CoM $x_G(\boldsymbol{q}) = \boldsymbol{f}_G(\boldsymbol{q}) \in \mathbb{R}^2$. Using a Jacobian $\boldsymbol{J}_G(\boldsymbol{q}) \in \mathbb{R}^{2\times 9}$,the second time derivative of CoM can be written as

$$\ddot{x}_G(\boldsymbol{q}) = \boldsymbol{F}_G(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{V}_G(\boldsymbol{q})\bar{\boldsymbol{u}}, \tag{5.15}$$

where

$$\boldsymbol{F}_G(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{J}_G\boldsymbol{M}^{-1}(-\bar{\boldsymbol{C}} + \boldsymbol{J}_c^{\mathrm{T}}\bar{\boldsymbol{\lambda}}) + \dot{\boldsymbol{J}}_G\dot{\boldsymbol{q}} \tag{5.16}$$

$$\boldsymbol{V}_G(\boldsymbol{q}) = \boldsymbol{J}_G\boldsymbol{M}^{-1}\bar{\boldsymbol{S}}. \tag{5.17}$$

So, when robot's CoM of the desired reference acceleration is determined as $\ddot{\boldsymbol{x}}_{Gd}$, the required input is calculated as

$$\bar{\boldsymbol{u}} = \boldsymbol{V}_G^{\dagger}(\ddot{\boldsymbol{x}}_{Gd} - \boldsymbol{F}_G), \tag{5.18}$$

where $\boldsymbol{V}_G^{\dagger}$ means a pseudo inverse of $\boldsymbol{V}_G$. This input works not only for realizing SLIP-like CoM movement,but also for other tasks. If we define all main tasks to be output function $\boldsymbol{x}_{t_1}$ as

$$\boldsymbol{x}_{t_1} = \boldsymbol{F}_{t_1}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{V}_{t_1}(\boldsymbol{q})\bar{\boldsymbol{u}}, \tag{5.19}$$

where $\boldsymbol{x}_{t_1} = [\boldsymbol{x}_{t_1,1}, \boldsymbol{x}_{t_1,2}, \cdots]^{\mathrm{T}}$ is the output vector and $\boldsymbol{x}_{t_1,i}$ is ith main task like the desired reference acceleration of CoM above. As considered in [31], CoM dynamics, centroidal angular momentum and position of ZMP are used for the main task.
In addition, there might be additional freedom of input, so we define a subtask to control the attitude of robot. This subtask also can be written as output function $\boldsymbol{x}_{t_2}$ given by

$$\boldsymbol{x}_{t_2} = \boldsymbol{F}_{t_2}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{V}_{t_2}(\boldsymbol{q})\bar{\boldsymbol{u}}, \tag{5.20}$$

where $\boldsymbol{x}_{t_2} = [\boldsymbol{x}_{t_2,1}, \boldsymbol{x}_{t_2,2}, \cdots]^{\mathrm{T}}$ is the output vector and $\boldsymbol{x}_{t_1,i}$ corresponds to ith sub task. In this work, we assign the defference between the angle of the floating leg's joint and the desired angle of it.
Using (5.19) and (5.20), the control input $\bar{\boldsymbol{u}}$ can be calculated as

$$\bar{\boldsymbol{u}} = \boldsymbol{V}_{t_1}^{\dagger}(\boldsymbol{x}_{t_1 d} - \boldsymbol{F}_{t_1}) + (\boldsymbol{I} - \boldsymbol{V}_{t_1}^{\dagger}\boldsymbol{V}_{t_1})\hat{\boldsymbol{V}}^{\dagger}(\boldsymbol{x}_{t_2 d} - \hat{\boldsymbol{F}}) \tag{5.21}$$

where $\boldsymbol{x}_{t_1d}$ is the desired value for the main task, $\boldsymbol{x}_{t_2d}$ is the desired value for the sub-task and

$$\hat{\boldsymbol{V}} = \boldsymbol{V}_{t_2}(\boldsymbol{I} - \boldsymbol{V}_{t_1}^{\dagger}\boldsymbol{V}_{t_1}), \tag{5.22}$$

$$\hat{\boldsymbol{F}} = \boldsymbol{V}_{t_2}\boldsymbol{V}_{t_1}^{\dagger}(\boldsymbol{x}_{t_1d} - \boldsymbol{F}_{t_1}) + \boldsymbol{F}_{t_2}. \tag{5.23}$$

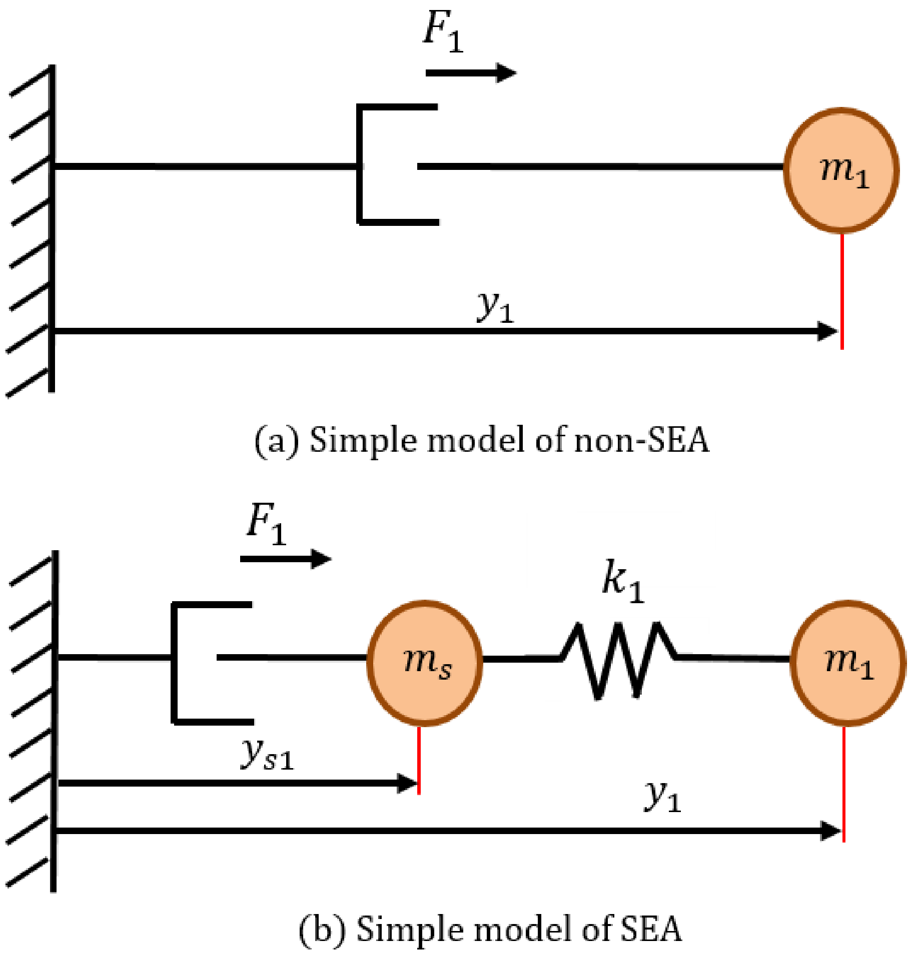


Figure 5.2: Simple model for analysis the non-SEA robot model (a) and SEA robot model (b). m_1, m_s and hatched area indicate the foot link, the link to drive SEA and lower leg link. In the mechanism (a), we can easily realize the desired acceleration $\ddot{y}_{1d}$ like GRF for imitating SLIP-like CoM movement. Otherwise, SEA mechanism like (b), calculation based on section III-A cannot be applied.

5.3.2 Difference between SEA and non-SEA model

The method to realize the SLIP-like CoM movement is based on control of the acceleration of CoM. To derive this control, the external forces should be taken into account such as gravity force and ground reaction force. Since the ground reaction force (GRF) is only controllable element to control the CoM acceleration, controllability of GRF from joint torques is important issue. In this section, we explain how the controllability of GRF from joint torques of SEA robot model differs from that of non-SEA robot model.

Fig. 5.2 shows a simplified model of the leg without the SEA in Fig. 5.2-(a) and with the SEA in Fig. 5.2-(b). m_1 indicates the foot link of the robot model, m_s indicates the link which drives the SEA element and the hatched area indicates the lower leg link. When the control law of the equation (5.21) is applied to the robot model, the acceleration of CoM is the output function for this controller. Likewise, in the system of Fig. 5.2, we can consider the acceleration of m_1 which is determined as $\ddot{y}_1$ as the output function. Suppose that the desired acceleration of m_1 is $\ddot{y}_{1d}$. In the case of Fig. 5.2-(a), this can be easily realized by a control input $\boldsymbol{F}_1 = m_1\ddot{y}_{1d}$. So, when the desired acceleration $\ddot{y}_{1d}$ which is imitating the SLIP-like CoM movement is specified, the desired torque for each joint can be calculated from the inverse dynamics of the system in Fig. 5.2-(a).

On the other hand, in the case of Fig. 5.2-(b), the acceleration of m_1 only depends on the difference between the length of spring and the natural length of it. So, when the desired acceleration output $\ddot{y}_{1d}$ is specified as same as the case of Fig. 5.2-(a), it is impossible to calculate directly the force $\boldsymbol{F}_1$ from the desired output function $\ddot{y}_{1d}$. To control the output function $\ddot{y}_1$, the force from the elasticity of $\tilde{y}_1 := y_1 - y_{s1}$ should be controlled properly. Likewise, to realize the desired acceleration of CoM in the SEA robot models, the method to calculate the control input mentioned above is not appropriate for the SEA models. So, in the next section, we propose a method to calculate the control input to realize the stable running for the SEA robot models.

5.3.3 Control law for SEA robot model

To realize the stable running gait for SEA robot models, imitating the joint acceleration of the running gait for the non-SEA robot model is adopted. Since there exist nine generalized coordinates in the non-SEA robot model, we need nine independent actuators to imitate all states in the SEA robot model to its non-SEA robot model. Unfortunately, we only have six actuators at each joint except the ankle joint, so we can only imitate six states of the non-SEA robot model. For imitating these six states, let us define the selected state vector $\bar{\boldsymbol{q}} \in \mathbb{R}^{6\times 1}$ as

$$\bar{\boldsymbol{q}} = \bar{\boldsymbol{W}}\boldsymbol{q}, \tag{5.24}$$

where $\bar{\boldsymbol{W}} \in \mathbb{R}^{6\times 9}$ is defined as

$$\bar{\boldsymbol{W}} = \begin{cases} \begin{bmatrix} \boldsymbol{I}_{6\times 6}, & \boldsymbol{0}_{6\times 3} \end{bmatrix} & (\mathit{other}) \\ \begin{bmatrix} \boldsymbol{0}_{6\times 1}, & \boldsymbol{I}_{6\times 6}, & \boldsymbol{0}_{6\times 2} \end{bmatrix} & (\mathit{sole\ contact}). \end{cases} \tag{5.25}$$

where $\boldsymbol{I}_{n\times n} \in \mathbb{R}^{n\times n}$ is an identity matrix, $\boldsymbol{0}_{i\times j} \in \mathbb{R}^{i\times j}$ is the matrix whose all elements are zero. From equations (5.14),(5.21), the desired joint acceleration is derived as

$$\ddot{\bar{\boldsymbol{q}}}_d = \bar{\boldsymbol{W}}\ddot{\boldsymbol{q}}_d = \bar{\boldsymbol{W}}\boldsymbol{M}^{-1}(-\bar{\boldsymbol{C}} + \bar{\boldsymbol{S}}\bar{\boldsymbol{u}} + \boldsymbol{J}_c^{\mathrm{T}}\bar{\boldsymbol{\lambda}}). \tag{5.26}$$

To utilize the spring force term, $\boldsymbol{C}$ is divided into two terms so that one represents the force from the spring elasticity at the ankle and the other represents remains. The reason why we consider about the elasticity is that they have relative degree 2 to the control input and it is larger than other terms.This can be written as

$$\boldsymbol{M}_a\ddot{\boldsymbol{q}}_a + \begin{bmatrix} \boldsymbol{C}_1' - \boldsymbol{f}_1(\boldsymbol{q}_a) \\ \boldsymbol{C}_2 \\ \boldsymbol{C}_3' - \boldsymbol{f}_2(\boldsymbol{q}_a) \\ \boldsymbol{C}_4 \end{bmatrix} = \boldsymbol{S}\boldsymbol{u} + \boldsymbol{J}_c(\boldsymbol{q})^{\mathrm{T}}\boldsymbol{\lambda}, \tag{5.27}$$

where $\boldsymbol{\theta}_o = [\theta_2, \theta_3, \theta_4, \theta_5]^{\mathrm{T}}$ is a generalized coordinate vector except θ_1 and θ_6, $C_1 \in \mathbb{R}^1, C_2 \in \mathbb{R}^4, C_3 \in \mathbb{R}^1, C_4 \in \mathbb{R}^5$ are the elements of $\boldsymbol{C}$ and $\boldsymbol{f}_i, i \in \{1, 2\}$ is the elastic force term in the ankle dynamics from the spring, $\boldsymbol{C}_i', i \in \{1, 3\}$ is defined as $\boldsymbol{C}_1' := \boldsymbol{C}_1 + \boldsymbol{f}_1, \boldsymbol{C}_3' := \boldsymbol{C}_3 + \boldsymbol{f}_2$. In the equation (5.27), by moving $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ to the right hand side, it can be rewritten as

$$\boldsymbol{M}_a\ddot{\boldsymbol{q}}_a + \underbrace{\begin{bmatrix} \boldsymbol{C}_1' \\ \boldsymbol{C}_2 \\ \boldsymbol{C}_3' \\ \boldsymbol{C}_4 \end{bmatrix}}_{:=\boldsymbol{C}'} = \underbrace{\begin{bmatrix} \boldsymbol{S} & \boldsymbol{S}' \end{bmatrix}}_{:=\boldsymbol{S}_a} \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{f} \end{bmatrix} + \boldsymbol{J}_c(\boldsymbol{q})^{\mathrm{T}}\boldsymbol{\lambda}, \tag{5.28}$$

where $\boldsymbol{f} := [\boldsymbol{f}_1, \boldsymbol{f}_2]^{\mathrm{T}}$ is the elastic force vector, $\boldsymbol{S}'$ is the matrix which is defined as

$$\boldsymbol{S}' := \begin{bmatrix} 1 & 0 \\ \mathbf{0}_{4\times1} & \mathbf{0}_{4\times1} \\ 0 & 1 \\ \mathbf{0}_{5\times1} & \mathbf{0}_{5\times1} \end{bmatrix}. \tag{5.29}$$

From equations (5.26),(5.28), the desired force term from the elasticity $\boldsymbol{f}_d \in \mathbb{R}^2$ should satisfy

$$\ddot{\bar{\boldsymbol{q}}}_d = \boldsymbol{W}\ddot{\boldsymbol{q}}_{ad} = \boldsymbol{W}\boldsymbol{M}_a^{-1}\left(-\boldsymbol{C}' + \boldsymbol{S}_a \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{f}_d \end{bmatrix} + \boldsymbol{J}_c^{\mathrm{T}}\boldsymbol{\lambda}\right), \tag{5.30}$$

where

$$\boldsymbol{W} = \begin{bmatrix} \bar{\boldsymbol{W}}, & \boldsymbol{0}_{6\times 2} \end{bmatrix}. \tag{5.31}$$

This notation shows that if the elastic force vector $\boldsymbol{f}$ could be controlled arbitrary, we can treat the system (5.28) so that it looks like there exist eight independent inputs which can also control the ankle joint arbitrary. However, the input vector $\boldsymbol{u}$ and the elastic force vector $\boldsymbol{f}$ are not independent, the dynamic relationship between these vectors can be written as

$$\dot{\boldsymbol{f}} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}_a}\dot{\boldsymbol{q}}_a \tag{5.32}$$

$$\ddot{\boldsymbol{f}} = \frac{\partial \dot{\boldsymbol{f}}}{\partial \boldsymbol{q}_a}\dot{\boldsymbol{q}}_a + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}_a}\ddot{\boldsymbol{q}}_a. \tag{5.33}$$

From the equation (5.4), $\ddot{\boldsymbol{q}}_a$ can be replaced by the function of $\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{u}$, then the equation (5.33) can be written as

$$\ddot{\boldsymbol{f}} = \boldsymbol{F}_f(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{V}_f(\boldsymbol{q})\boldsymbol{u}, \tag{5.34}$$

where

$$\boldsymbol{F}_f(\boldsymbol{q}_a, \dot{\boldsymbol{q}}_a) := \frac{\partial \dot{\boldsymbol{f}}}{\partial \boldsymbol{q}_a}\dot{\boldsymbol{q}}_a + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}_a}\boldsymbol{M}_a^{-1}(-\boldsymbol{C} + \boldsymbol{J}_c^{\mathrm{T}}\boldsymbol{\lambda}) \tag{5.35}$$

$$\boldsymbol{V}_f(\boldsymbol{q}_a) := \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}_a}\boldsymbol{M}_a^{-1}\boldsymbol{S}. \tag{5.36}$$

5.4 Simulation result

Using the proposed method in section 5.3, we conduct a running simulation when a desired acceleration is determined by the non-SEA model described in section 5.2. The reference running velocity is set to 2 [m/s] and desired jumping height of CoM is set to 0.97 [m]. Control parameters in section 5.3 are set to $k_0 = 250000$ and $k_1 = 1000$. Figure 5.3 shows the phase diagram of CoM when the robot's running is stable. The phase diagram of the stable running of the non-SEA model is expressed by the blue line and the phase diagram of the CoM of the SEA robot model is expressed by the red line. Dashed line shows the zero velocity of CoM in the vertical direction, so the point which crosses the dashed line on the phase plot in left-side indicates the state of the lowest position in the stance phase, and the point which crosses the dashed line on the phase plot in right-side indicates the state of the highest position in the flight phase. This figure shows that there is small differences between the phase plot of non-SEA model and that of SEA model due to the approximated method.

Fig. 5.4 shows the tracking performance of the force term of the elasticity. The desired force of elasticity for tracking the acceleration of each joint is calculated from equation (5.26) and it is expressed as red line in Fig. 5.4. Blue line indicates the actual force term of the elasticity. This tracking ability depends on the control parameters, k_0 and k_1,which assign the poles of the tracking error dynamics. When the large values for stable poles are assigned, tracking ability will be improved but robust stability for uncertainties might be deteriorated, on the contrary, the small values for stable poles are assigned, tracking ability becomes worse but robust stability might be improved. If the poles are too small, running gait of the robot will be unstable.

Fig. 5.5 shows the actual and the desired vertical CoM acceleration of the running gait in the stance phase. Blue line shows the desired vertical CoM acceleration to realize the SLIP-like CoM movement. Red line shows the actual CoM acceleration which is generated by the force from the SEA. Dashed line shows an instance when a feedback for energy compensation is started. In the case of the non-SEA robot model, there is no error between the desired and the actual accelerations because the acceleration of CoM can be calculated by the desired one directly. On the other hand, in the case of the SEA robot model, the acceleration of CoM has some phase lag due to the dynamics between the elastic force and the SEA's motor torque as described in section 5.3.2.

5.5 Conclusion

In this chapter, we presented how to control the SEA robot models for realizing the stable running gaits based on control methods for non-SEA robot model. We stated why the control laws for the non-SEA robot models can't be applied to the SEA robot models directly and proposed how to calculate the actuator inputs for realizing that the CoM of SEA robot models imitates the SLIP model's stable running and desired accelerations determined by non-SEA models. To realize the desired torque at the ankle joint, we used the force term of elasticity which has a relative degree 2 to the control input. The proposed control input may not be optimal for SEA robot models, however, the proposed method realized stable running based on knowledge about control methods for non-SEA robot models.

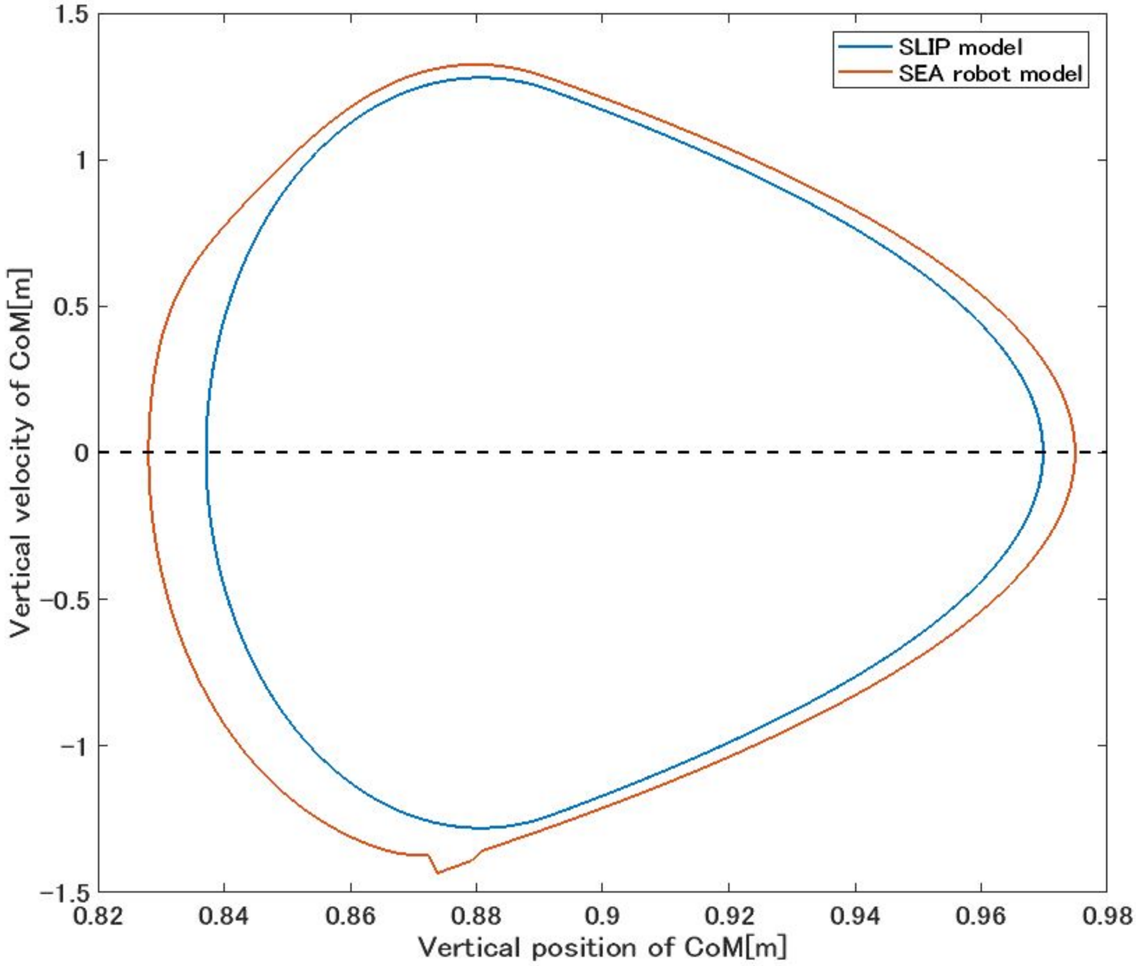


Figure 5.3: This figure shows the phase diagram of CoM when the robot's running is stable. Blue line indicates the phase of stable running of the non-SEA model, Red line indicates the phase of stable running of the SEA robot model and dashed line indicates the zero velocity of CoM in vertical direction. Cross point with left-side is the lowest position in stance phase and the right-side of it is the highest position in flight phase.

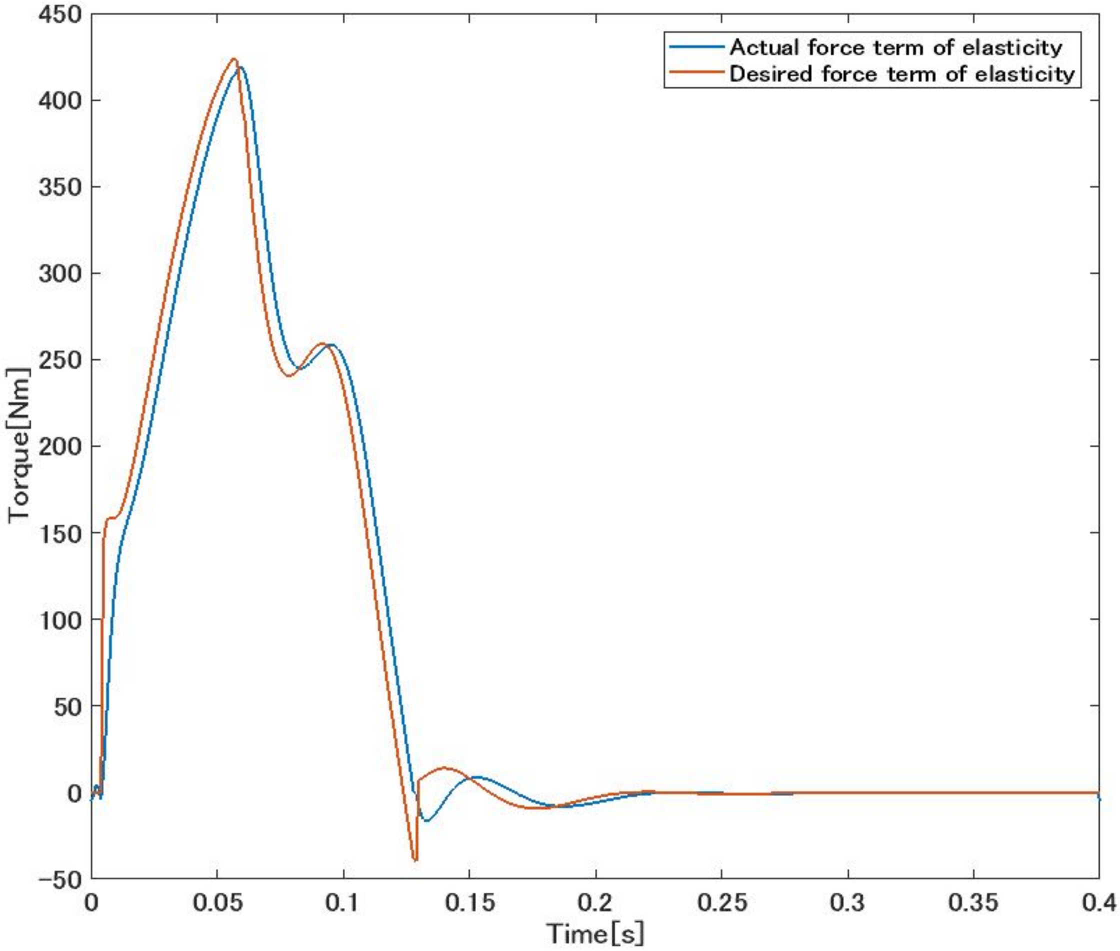


Figure 5.4: Tracking the force term of spring elasticity. Desired and actual force term of elasticity are indicated by red and blue line. Tracking ability depends on the poles of tracking error dynamics which are determined by the control parameters k_0 and k_1.

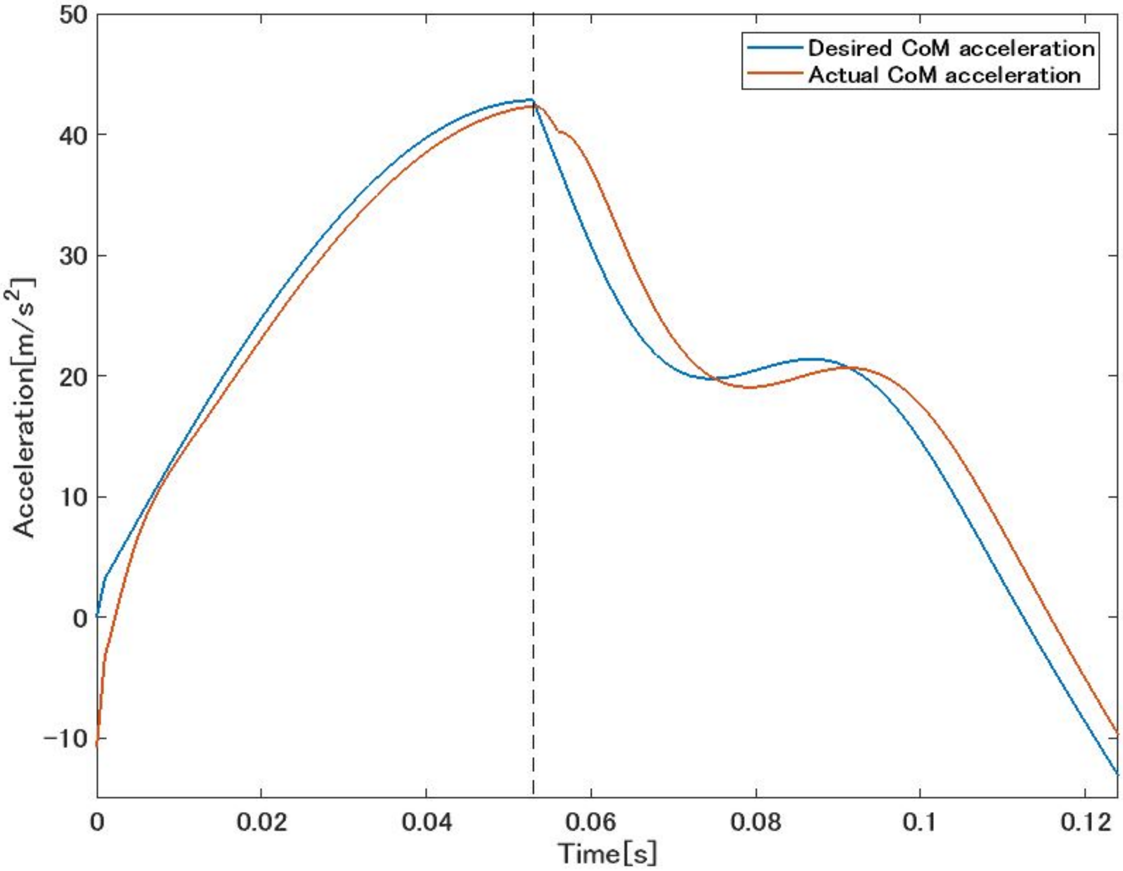


Figure 5.5: The actual and desired vertical CoM acceleration in stance phase. The error appears due to the SEA mechanism. The feedback for the energy conservation is generated after the time which is indicated by dashed line. This error depends on the control parameter k_0 and k_1.

Chapter 6

Verification of mechanical improvement for bipedal walking robot

6.1 Introduction

Recent studies have proposed methods for improving not only control techniques but also mechanical structures in order to improve the walking performance of biped robots. Since the mechanical structure of a robot is an important property for locomotion when performing a task given to the robot and the natural dynamics of the robot itself, it greatly affects the efficiency and stability of the robot. So, in recent researches, various mechanical structures for bipedal robots have been proposed to improve the performance of the robot's walking such as equipping an elasticity element and adjusting inertial properties to the joints of the robots.

Adding the elastic elements to rigid dynamics of the robot is an important feature of legged robot's walking [34] [35]. The elastic property of the legged robot has several advantages in performing tasks, such as storing the elastic energy by compliance elements to improve the efficiency and improving the tolerance of the shock from the environment [36]. Recent studies using this elastic properties have focused on optimal implementation of the elastic elements to perform their own tasks optimally. In [37] [38], the additional actuators for adjusting the stiffness of the elasticity have been introduced to optimize the dynamic properties for tracking the desired trajectory of the manipulator. In [39], the mechanical design of the robot including the passive compliance was determined to perform walking optimally evaluating with joint torques and walking stability. However, since the required elastic properties differ depending on the task given to the robot, it is considered that the optimal elastic stiffness differs depending on the behavior even for the same robot.

As another mechanical structure for improving the performance of the robot's walking, a mechanism for adjusting the moment of inertia of the joints, which is called "inerter", and a mechanism for considering the effect of the arm swinging motion of a walking person on the moment of inertia of the upper body , which is called "wobbling mass", have been proposed in the previous researches. The inerter was introduced as a linear motion model in [40], and the rotational version for mounting on the joints of the bipedal robot was proposed in [41]. This mechanism improved the performance of the passive walker on the slope whose performance was evaluated with the maximum speed and the energy efficiency of the gait.

On the other hand, the wobbling mass was also introduced for improving the performance of the walking gait [42]. The motivation of this mechanism is based on the fact that the up-and-down movement of the mass of the robot improves the performance of the legged walking such as easing the peak vertical force [43] and increasing the walking speed of the natural dynamics of a legged robot [44]. These motivations are based on the assumption that the arm swinging movement in the human's walking improves the performance of the gait by moving the center of the upper body's mass up-and down, and in [42], mimicking this movement with the wobbling mass mechanism's was successfully implemented for the gait of the limit cycle walker by increasing the walking speed of the robot. From this result, we expect that devising this mechanism to the bipedal robot could improve the performance of the walking by optimizing the parameters for these mechanical structures.

In this chapter, we present the verification of mechanical improvement for bipedal walking robots with adding series elasticity and mechanical elements to the joint structure including inerter and wobbling mass. In particular, we used the fully-actuated five link robot as a nominal model, and we devised the series elastic actuator and inerter on the hip joint and the knee joint of the robot, respectively. The wobbling mass is implemented in the upper body as a mass and a linear actuator.

6.2 Target systems and model

6.2.1 Implemented mechanical structure

Series elastic actuator

In this chapter, we use the dynamics of the series elastic actuator as follows. Since we are focusing on the effect of the synergy of mechanical structures in the gait of the bipedal robot, we assume that these elements consist of a simple structure of each element as follows. The mechanical schematic of the structure of series elastic actuator is shown in Figure 6.1. In this work, we assume the inertial coupling between the motor's link and the joint link can be omitted for simplicity of the model. From this assumption, the dynamics of the series elastic actuator is independent of the joint acceleration, and can be written as

$$\begin{aligned} \boldsymbol{B}\ddot{\boldsymbol{\phi}} + \boldsymbol{\tau} &= \boldsymbol{u} && (6.1)\\ \boldsymbol{\tau} &= \boldsymbol{R}(\dot{\boldsymbol{\phi}} - \dot{\boldsymbol{\theta}}) + \boldsymbol{K}\left(\boldsymbol{\phi} - \boldsymbol{\theta}\right), && (6.2) \end{aligned}$$

where $\boldsymbol{\phi}$ is the angle of the motor, $\boldsymbol{\theta}$ is the relative angles of the links, $\boldsymbol{B} = diag(b_i)$ is an inertia matrix of the motors, $\boldsymbol{R} = diag(r_i)$ is a damping matrix of the spring, $\boldsymbol{K} = diag(k_i)$ is a stiffness matrix of the spring, and $\boldsymbol{u}$ and $\boldsymbol{\tau}$ are the input torque from the motor and torque of the series elasticity, respectively. In this work, the damping coefficient r_i and the spring stiffness k_i was assigned as

$$k_i := b_i \cdot p_i^2, \qquad r_i := 2 \cdot b_i \cdot p_i, \tag{6.3}$$

where p_i is the pole of the spring-mass-damper system, so that the natural dynamics of the series elastic actuator doesn't have the oscillation.

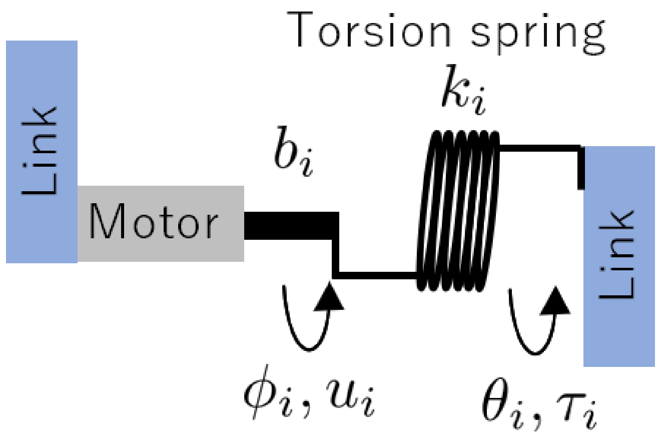


Figure 6.1: Mechanism of series elastic actuator

Inerter

The dynamics of the inerter and wobbling mass is described as follows. In this work, we used the mechanism of the rotational inerter which was proposed by Hanazawa [41], which is shown in Figure 6.2. This mechanism increases the momentum of the joint with the gear, pinion, and the flywheel elements but does not change the mass. The dynamics model of rotational inerter is shown in Figure 6.3, and the dynamic equation of this model can be written as

$$\tau_I = \beta\ddot{\theta}_i, \tag{6.4}$$

where $\beta := J_f\alpha_1^2\alpha_2^2, \alpha_1 := r_{g1}/r_{p1}$ and $\alpha_2 := r_{g2}/r_{p2}$ is the momentum of the joint, and θ_i indicates the relative angle of the joint devised with this mechanism.

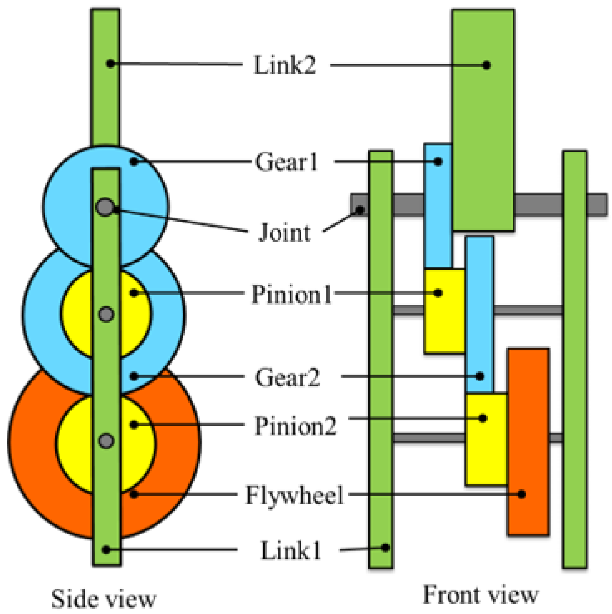


Figure 6.2: Mechanism of inerter

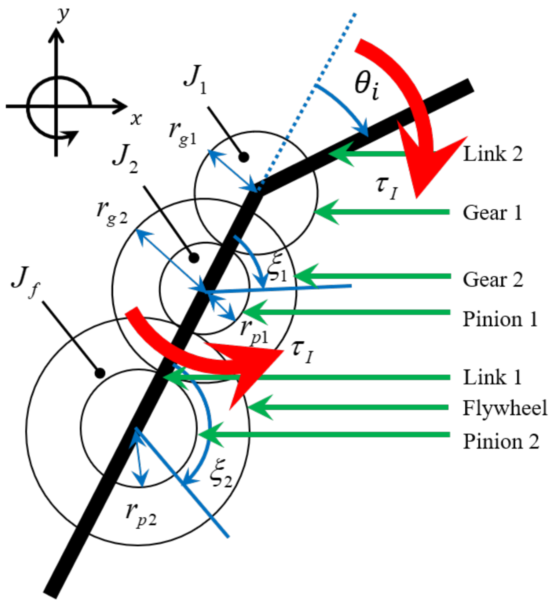


Figure 6.3: Dynamics model of inerter

Wobbling mass

The mechanism of the wobbling mass which was proposed in [42] is shown in Figure 6.4. Since this device simulates arm swinging by moving the wobbling mass up-and-down, the additional linear actuator is considered in the upper body of the biped robot.

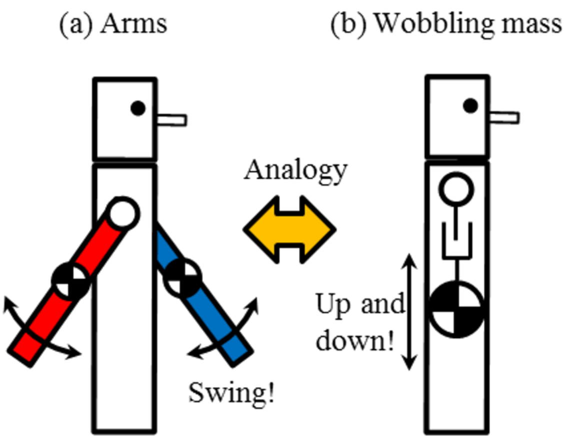


Figure 6.4: Mechanism of wobbling mass

6.2.2 Biped robot model

In this section, we describe the derivation of mathematical model of biped robot to verify the effectiveness of mechanical improvement. The nominal robot model's parameters were assumed to be same as the robot model in [45]. The robot model in [45] has only four actuators, so that the original version is supposed as under-actuated robot, but we assume that there exists additional actuator on the stance foot's ankle which can be supposed as fully-actuated robot model.

Dynamics of the conventional robot model

First, we describe the dynamics of conventional biped walking robot model as shown in Figure 6.5. This model is assumed to be a fully-actuated, which consists of five rigid bodies and five actuators to control. The dynamics of this model can be written as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{S}\boldsymbol{u} + \boldsymbol{J}_c^T\boldsymbol{\lambda}, \tag{6.5}$$

where $\boldsymbol{q} = [\theta_c, \theta_1, \theta_2, \theta_3, \theta_4, x_c, y_c]^T \in \mathbb{R}^7$ is a generalized coordinate vector of the system, $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{7\times7}$ is an inertia matrix, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{7\times7}$ is a Coriolis and centrifugal force matrix, $\boldsymbol{G}(\boldsymbol{q}) \in \mathbb{R}^7$ is a gravitational force vector, $\boldsymbol{S} \in \mathbb{R}^{7\times5}$ is a transformation matrix, and $\boldsymbol{u} \in \mathbb{R}^5$ is a input vector, and $\boldsymbol{J}_c^T\boldsymbol{\lambda}$ indicates the constraint force on stance leg's foot caused by the reaction from the ground. This constraint force can be calculated as shown in section 2.1.2, so detail of this will be omitted.

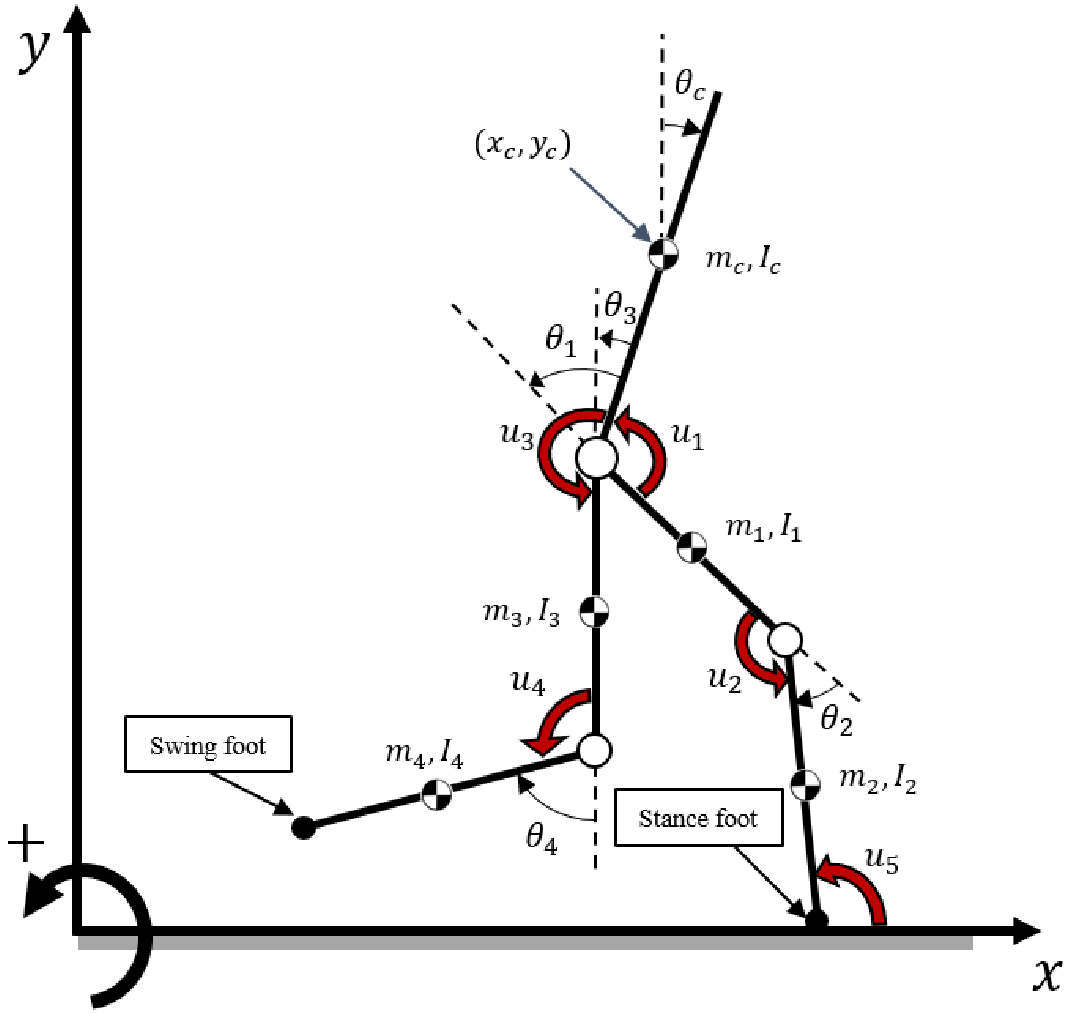


Figure 6.5: 5 link robot model

Table 6.1: 5 link model parameters

m_c : 12.0 [kg]	$m_1 = m_3$: 6.8 [kg]	$m_2 = m_4$: 3.2 [kg]
l_c : 0.625 [m]	$l_1 = l_3$: 0.4 [m]	$l_2 = l_4$: 0.4 [m]
I_c : 1.33 [kg$\cdot$ m^2]	$I_1 = I_3$: 0.47 [kg$\cdot$ m^2]	$I_2 = I_4$: 0.20 [kg$\cdot$ m^2]

In this walking gait, we assume that the state transition by the collision with the ground occurs instantly, and in the instant of the collision, we assume that the stance leg foot is also off the ground instantaneously. The robot dynamics is assumed to be symmetric for the left and right sides of the robot, so that the one period of the template model is from the swing leg takeoff event to the swing leg ground contact event. From this assumption, whole system of the gait can be expressed by combination of 1 continuous system which represents the single stance phase and 1 discrete system which represents collision with the ground.

In this work, we used the robot's parameter as presented in [45]. The mass, the momentum of inertia, and the length of the each rigid body is described in Table 6.1.

Dynamics of the improved robot model

In this section, we describe the mechanically devised robot model with series elastic actuator, inerter, and wobbling mass as shown in the Figure 6.6. This model consists of five rigid links, a point mass in the upper body, and the five inputs where four of them are implemented on each joint, and one is on the upper body. The inerter is devised on the knee joint, indicated on the knee joint as yellow. The series elastic actuator is devised on the hip joint whose torque is indicated as a blue arrow. The wobbling mass is implemented in the upper body, which actuate a point mass up-and-down. The dynamics of the whole system can be written as

$$\begin{bmatrix} \boldsymbol{M}(\boldsymbol{q}) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{B} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}} \\ \ddot{\boldsymbol{\phi}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) - \boldsymbol{\tau} \\ \boldsymbol{\tau} \end{bmatrix} = \begin{bmatrix} \boldsymbol{u}_a + \boldsymbol{J}_c^T \boldsymbol{\lambda} \\ \boldsymbol{u}_b \end{bmatrix}, \tag{6.6}$$

where $\boldsymbol{q} := [\theta_c, x_w, \theta_1, \theta_2, \theta_3, \theta_4]^T$ is the state vector and x_w represents the position of the wobbling mass, $\boldsymbol{\phi} := [\phi_1, \phi_2]^T$ represents the angle of the rotor of series elastic actuator, $\tau \in \mathbb{R}^2$ is the torque vector applied on the hip joint actuated by the series elastic actuator, $\boldsymbol{u}_a = [u_1, u_3, u_5, u_6]^T \in \mathbb{R}^4$ is the input vector contains the actuator of the wobbling mass and the actuators of the knee and the ankle joint, $\boldsymbol{u}_b = [u_2, u_4]^T \in \mathbb{R}^2$ is another input vector to actuate the series elastic element in the hip joint. We set the parameters for this model as shown in Table 6.2.

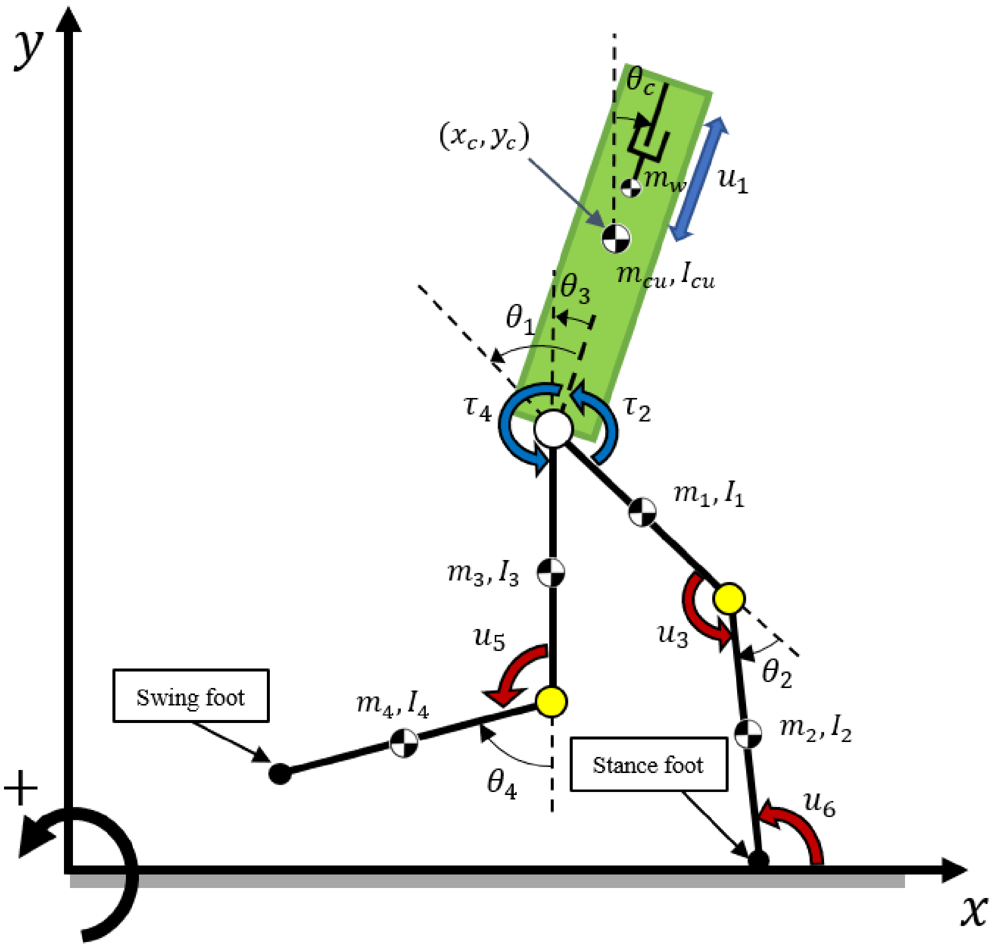


Figure 6.6: Devised 5 link robot model

Table 6.2: Devised robot model parameters

m_{cu} : 11.5 [kg]
m_w : 0.5 [kg]
b_i : 0.005 [kg $\cdot$ m^2] where $i \in \{1, 2\}$

6.3 Optimization method

In this section, we implemented the optimization method to find the optimal value of the parameters of the mechanical devices such as the elastic stiffness of the series elastic actuator, the moment of inertia of the inerter, and the optimized trajectory of the movement of the wobbling mass. In this work, we used the nonlinear programming solver to solve the optimization problem with OpenOCL as described in section 3.3.

The optimization problem can be formulated as:

$$\begin{aligned} \min_{\boldsymbol{x},\boldsymbol{p},\boldsymbol{u},T} \quad & \int_0^T J(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p})dt \\ \text{s.t.} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{p}) \\ & \boldsymbol{r}_k(\boldsymbol{x}(\mu_k), \boldsymbol{p}) \leq 0, \end{aligned} \tag{6.7}$$

where $t \in [0, T]$ is the time, T is the end time of the stance phase, $\boldsymbol{x}(t)$ is the state vector, $\boldsymbol{p}$ is the parameter vector, $J(x, p)$ is the path cost function, $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{p})$ represents the dynamics function of the system which is represented as differential equation and $\boldsymbol{r}_k(\boldsymbol{x}, \boldsymbol{p})$ is the grid-constraints function. In this study, the Cost of Transport (CoT) [9] which indicates the efficiency of the walking gait was used as same in section 3.3.

We set the path, boundary constraints to find a feasible solution as follows:

Path constraints

- Boundary of the parameters were assigned to be:

$$\begin{cases} p_1 = p_2 = p^* \\ 0 \leq \beta_1 = \beta_2 \leq \beta_{max}, \end{cases} \tag{6.8}$$

  where $p_i, \forall i \in \{1, 2\}$ are the poles of the spring-mass-damper system of the series elastic actuators, and $\beta_i, \forall i \in \{1, 2\}$ are the momentum of the inerter device on the knee joints.

- The vertical position of the swing foot always greater than 0:

$$P_{foot,y}(\boldsymbol{\theta}) \geq 0. \tag{6.9}$$

- The knee joint doesn't bend to the other side:

$$\theta_2 \leq 0, \qquad \theta_4 \leq 0. \tag{6.10}$$

- The upper body doesn't tilt backward or too much for forward:

$$-\theta_{c,max} \leq \theta_c \leq 0, \tag{6.11}$$

  where $\theta_{c,max}$ represent the upper bound of the tilt angle of the upper body. Negative direction means the tilt to the forward.

Boundary constraints

- The initial state $\boldsymbol{q}(t = 0)$ and the final state $\boldsymbol{q}(t = t_f)$ of the periodic movement have following relation :

$$\boldsymbol{q}(0) = \Delta(\boldsymbol{q}(t_f)), \tag{6.12}$$

  where Δ is a reset map of the joint angle.

- The state transition of the velocity vector is occurred by the impact event, and this can be described as:

$$\dot{\boldsymbol{q}}^{+}(0) = \Delta(\dot{\boldsymbol{q}}^{+}(t_f)), \tag{6.13}$$

  where, $\dot{\boldsymbol{q}}^{+}(t_f)$ represent the state after the impact event, which is calculated by the state before the impact event $\dot{\boldsymbol{q}}^{-}(t_f)$ as described in Section 2.1.3.

- The initial and the final position of the swing leg were assign to be:

$$P_{foot,x}(\boldsymbol{\theta}(0)) = -L_{step}, \quad P_{foot,x}(\boldsymbol{\theta}(t_f)) = L_{step}, \tag{6.14}$$

  where the L_{step} is the step length of each simulation.

- The swing leg should touch the ground within a set time:

$$t_f = T. \tag{6.15}$$

6.4 Optimization result

In this section, we describe the optimized trajectory and parameters of the nominal five link model and the improved model to confirm the effect of the devices such as series elastic actuators, inerter and wobbling mass on the energy efficiency of the walking gait. To compare this, we used Cost of Transport [9] as an indicator of the walking efficiency as described in Section 3.3. And, for the parameters of the optimization setting, we chose the following values as described in Table 6.3.

Table 6.3: Optimization parameters

$p^* : 1000$	$\beta_{max} : 0.1$ [Nm/(rad/s^2)]
$\theta_{c,max} : 0.2$ [rad]	$L_{step} : 0.2$ [m]
$T : 0.4$ [sec]	

Using the above setting, the optimal walking gaits were obtained by solving the optimization problem. The walking speed was set to 0.45 [m/s], and the optimization was implemented with three different models, the nominal five link model (*Case 1),* the five link model that only the series elastic actuators is devised (*Case 2),* and the five link model that was applied both the series elastic actuators and the inertial devices including the inerter and the wobbling mass (*Case 3).*

Table 6.4: Optimized parameters of mechanical devices

	Case 1	Case 2	Case 3
Stiffness of Series Elasticity [Nm/rad]	-	3.2265×10^3	3.2547×10^3
Inertial Momentum of Inerter [Nm/(rad/s^2)]	-	-	1.00×10^{-3}

Figure 6.7 shows the optimized gait of the nominal five link model and Figure 6.8 shows the optimized gait of the improved robot model. The results of the optimized input torque and the states of the nominal five link model are shown in Figure 6.9 ˜6.11. Figure 6.12 ˜6.15 show the results of the model only with the series elastic actuators, and Figure 6.16 ˜6.18 show the optimized results of the improved model with the series elastic actuators, inerter, and wobbling mass. The optimized parameters are given as Table 6.4 and Figure 6.20 shows the optimized trajectory of the wobbling mass in the gait of the improved model. The maximum amplitude of the input torque of the nominal model was obtained as 37.57 [N/m], where as the model with series elastic actuators was obtained as 33.57 [N/m], and in the model with whole devices case, it was obtained as 25.88 [N/m].

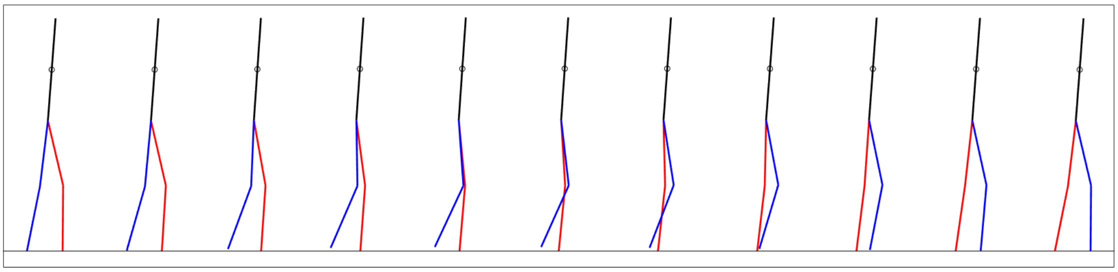


Figure 6.7: Walking gait of the nominal five link robot model

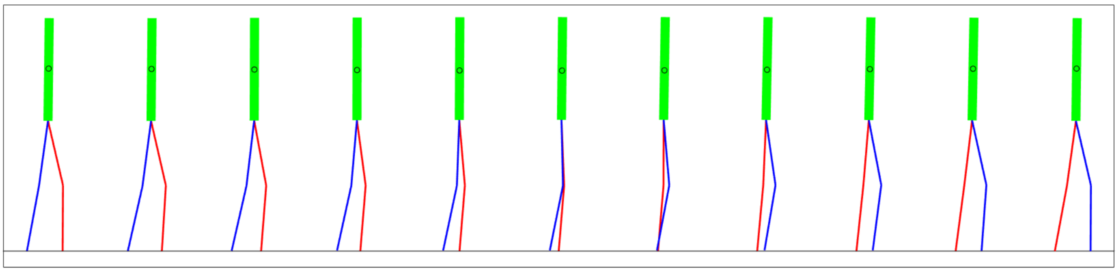


Figure 6.8: Walking gait of the improved robot model

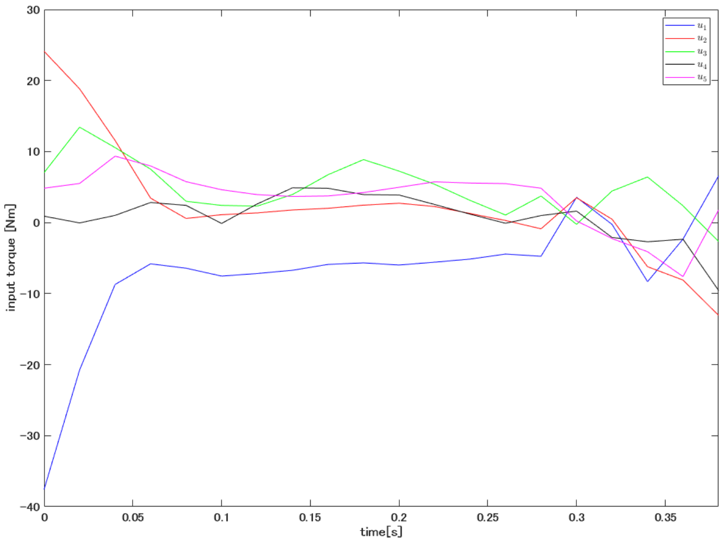


Figure 6.9: Optimized input for the nominal five link model

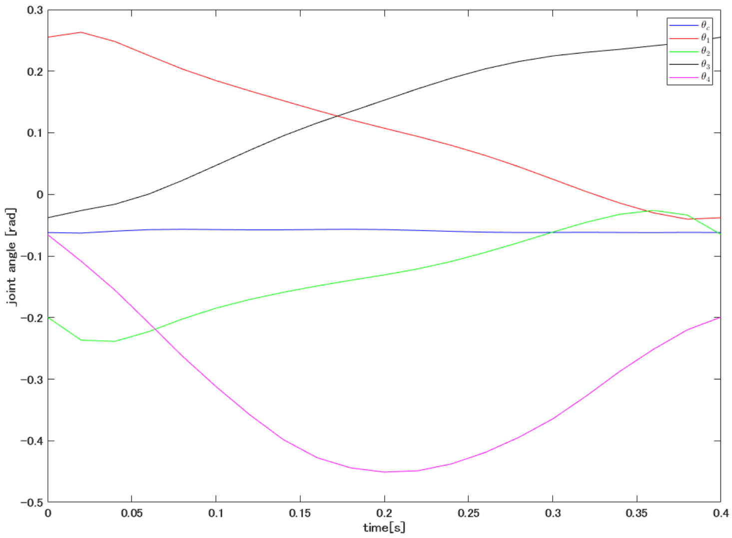


Figure 6.10: Optimized joint angle for the nominal five link model

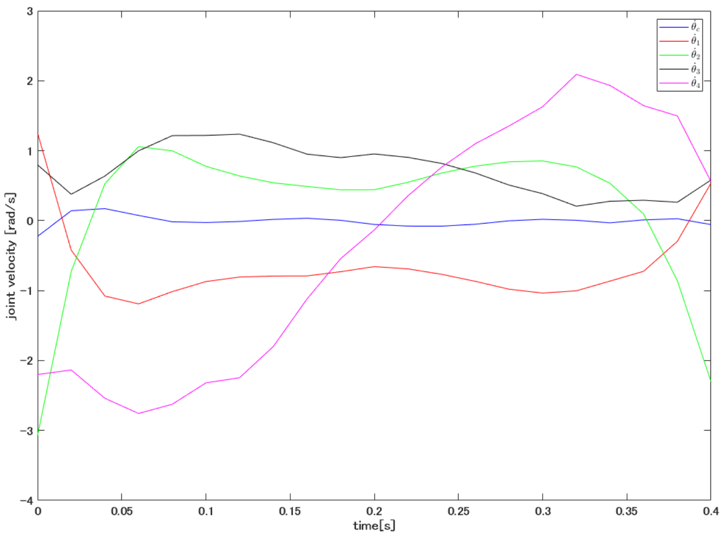


Figure 6.11: Optimized joint velocity for the nominal five link model

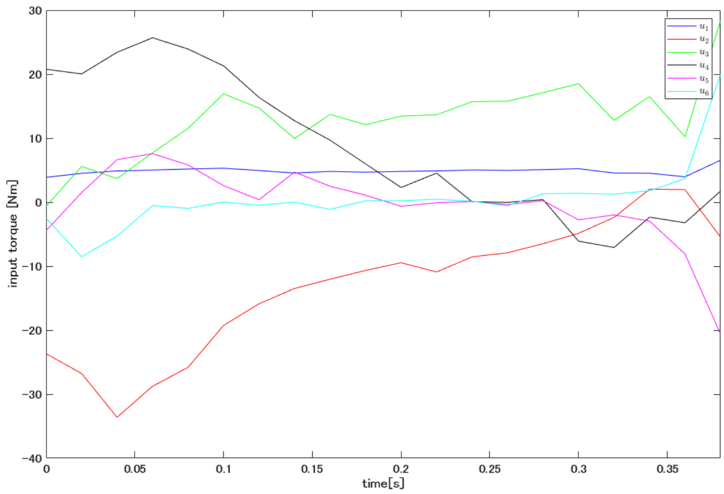


Figure 6.12: Optimized input for the robot model with SEA

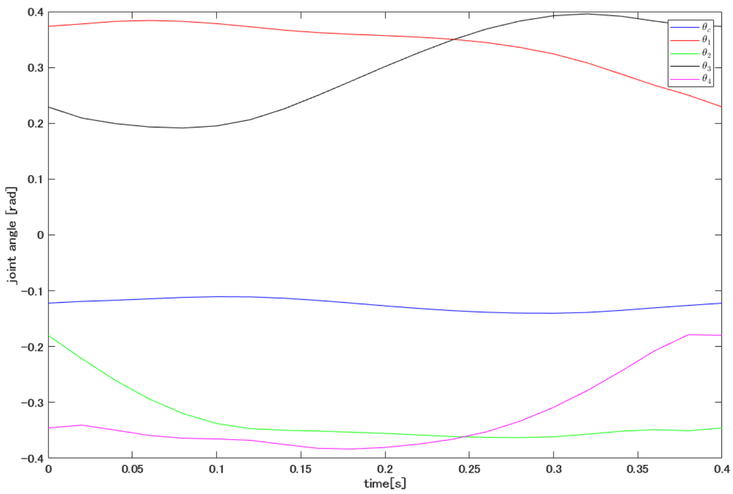


Figure 6.13: Optimized joint angle for the robot model with SEA

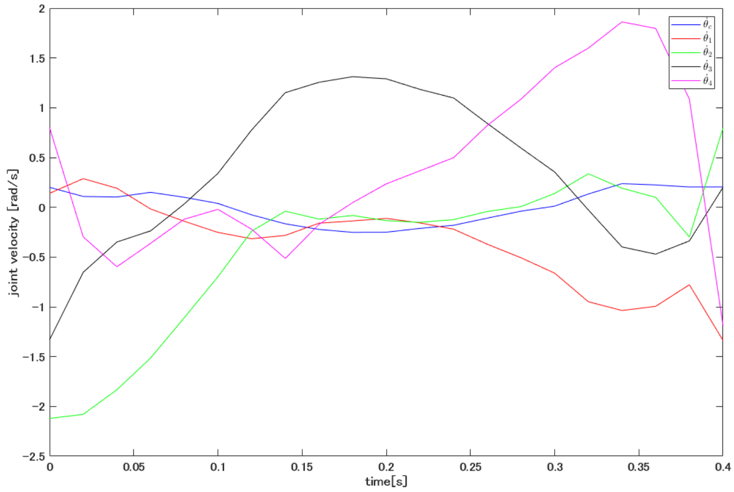


Figure 6.14: Optimized joint velocity for the robot model with SEA

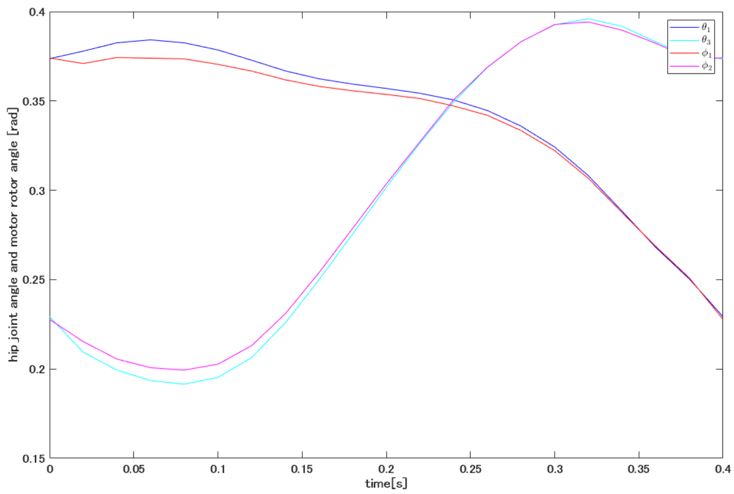


Figure 6.15: Optimized motor rotor angle for the robot model with SEA

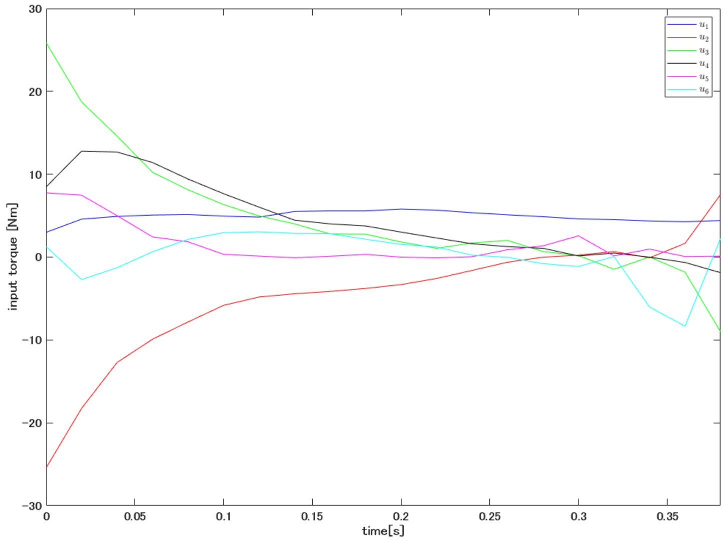


Figure 6.16: Optimized input for the improved robot model

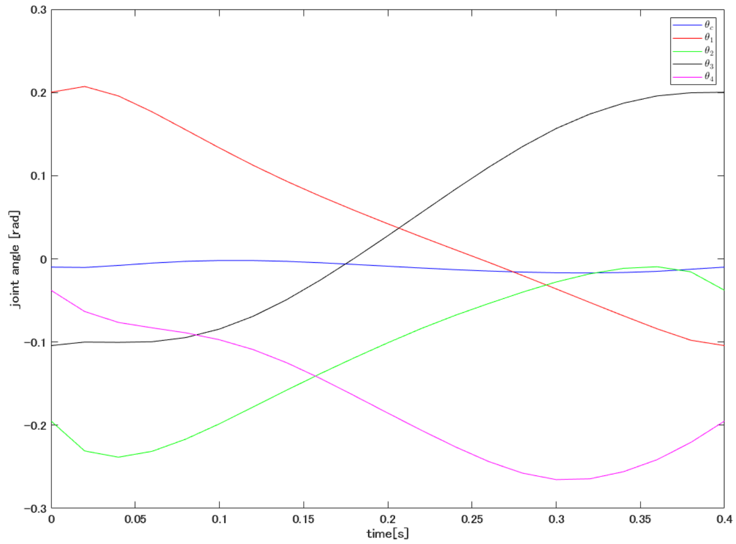


Figure 6.17: Optimized joint angle for the improved robot model

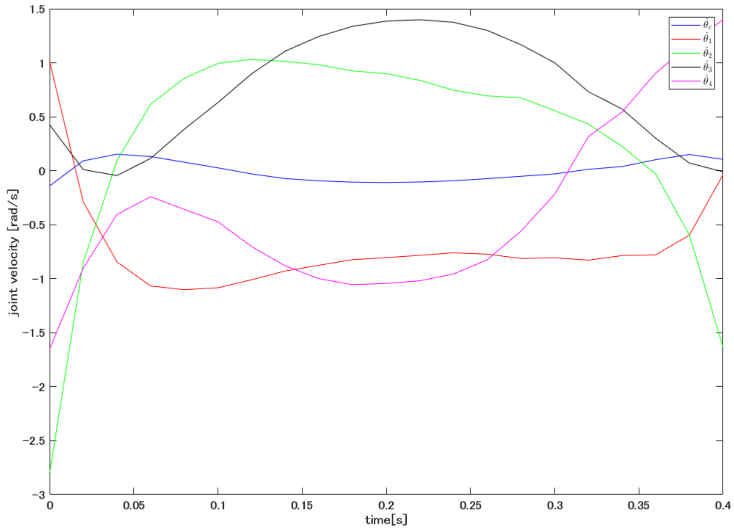


Figure 6.18: Optimized joint velocity for the improved robot model

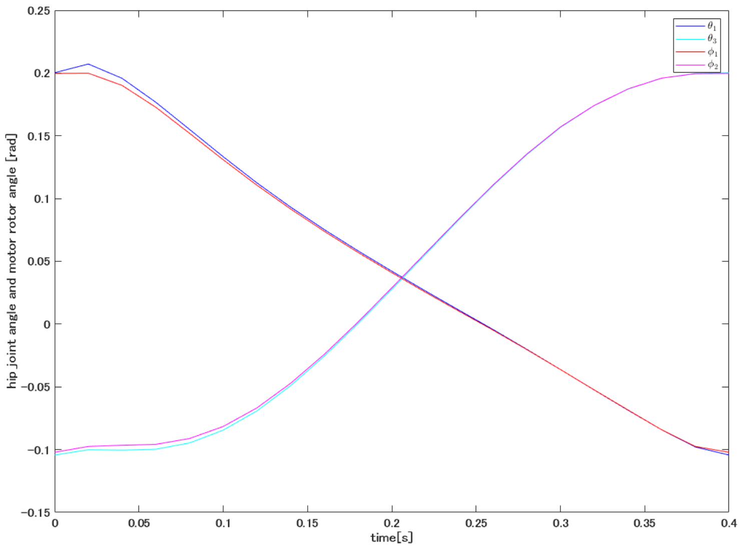


Figure 6.19: Optimized motor rotor angle for the improved robot model

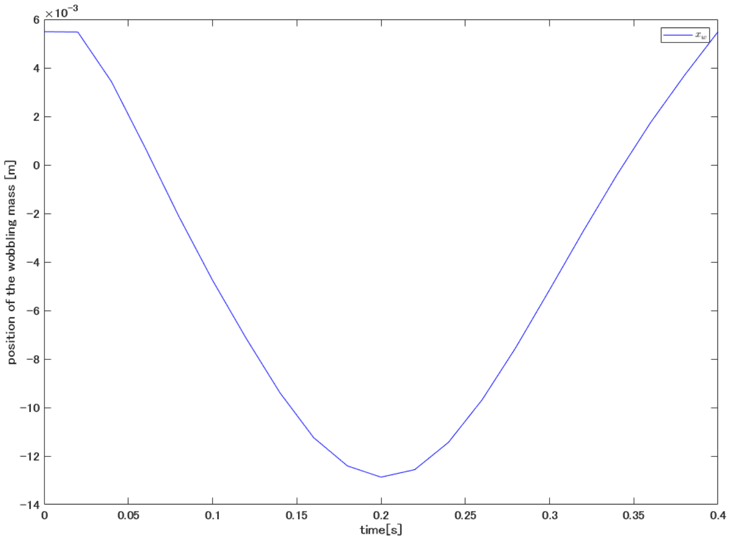


Figure 6.20: Optimized trajectory of the wobbling mass

The CoT value of the nominal five link model's gait was 0.0803, the model only with the series elastic actuators was 0.0706, and the model that was applied both the series elastic actuators and the inertial devices including the inerter and the wobbling mass was 0.0543. Compared with the results of the nominal model case, 12.08% improvement of energy efficiency was observed in the case of the model only with the series elastic actuators, and 32.38% improvement of energy efficiency was observed in the case of the model with both devices.

The optimized inertial momentum of inerter device was obtained at the boundary of the parameters, so in order to verify that this is the optimal value for the inerter, we also confirmed the optimization results in the range $\beta_{min} = 0$ [Nm/(rad/s^2)] and $\beta_{max} = 0.001$ [Nm/(rad/s^2)] by setting the initial conditions as a result of the optimized parameters and the trajectory of the improved model. In this setting, we obtained the optimal parameter for the inerter as $\beta = 6.85 \times 10^{-4}$ [Nm/(rad/s^2)], and the CoT value of this gait was 0.0727. From this result, we concluded that the optimal value for the inerter parameter is $\beta = 1.00 \times 10^{-3}$ [Nm/(rad/s^2)].

6.5 Control method

In this work, we used output zeroing control to stabilize the walking of the devised five link model. The reference output functions need to be continuous functions with respect to the time domain, but the optimized trajectory obtained from the previous section is a set of discretized states, so it is necessary to interpolate the discretized states in order to apply it to the continuous time system. In this work, we approximated using a seventh order Bézier curve to make continuous output functions with the discretized states. The seventh order Bézier curve can be written as

$$B_{s,i}(s) := \sum_{k=0}^{7} P_{k,i} \frac{7!}{k!\,(7-k)!} s^k (1-s)^{7-k}. \tag{6.16}$$

where $s \in [0,1]$, and $i \in \{\theta_c, x_w, \theta_2, \theta_4, \phi_1, \phi_2\}$ represents the joint of the model. The domain of variable s is defined in 0 to 1 whereas the domain of the optimized trajectory t is in 0 to 0.4 [s], we assume that $s = \kappa(t)$ where function κ is a class $\mathcal{K}$-function. To decide the control points $P_{k,i}$ of the Bézier curve, the following conditions are considered:

- The initial position $q(0)$ and the final position $q(T)$ of the optimized trajectory are satisfying

$$P_{0,i} = q_i(0), P_{7,i} = q_i(T) \tag{6.17}$$

- The initial velocity $\dot{q}(0)$ and the final velocity $\dot{q}(0)$ of the optimized trajectory are satisfying

$$P_{1,i} = \frac{1}{n}\frac{\partial t}{\partial s}\dot{q}_i(0) + P_{0,i}, P_{6,i} = -\frac{1}{n}\frac{\partial t}{\partial s}\dot{q}_i(0) + P_{7,i} \tag{6.18}$$

- The remaining control points $P_{2,i}, P_{3,i}, P_{4,i}, P_{5,i}$ were decided as follows

$$\begin{bmatrix} P_{2,i} \\ P_{3,i} \\ P_{4,i} \\ P_{5,i} \end{bmatrix} = A^{\dagger} \begin{bmatrix} q_i^*(\Delta t) - c_i(\Delta t) \\ q_i^*(2\Delta t) - c_i(2\Delta t) \\ \vdots \\ q_i^*(T-\Delta t) - c_i(T-\Delta t) \end{bmatrix}, \tag{6.19}$$

where $q_i^*(\cdot)$ is the optimized discrete state, Δt is the interval of discrete state, $c_i(\cdot)$ is defined as

$$\begin{aligned} c_i(t) \ := \ & P_{0,i}(\kappa(t))^7 + 7P_{1,i}\kappa(t)\left(1-\kappa(t)\right)^6 \\ & +7P_{6,i}(\kappa(t))^6\left(1-\kappa(t)\right) + P_{7,i}\left(1-\kappa(t)\right)^7, \end{aligned} \tag{6.20}$$

and A is defined as

$$A := \begin{bmatrix} a_{2,i}(\Delta t) & a_{3,i}(\Delta t) & a_{4,i}(\Delta t) & a_{5,i}(\Delta t) \\ a_{2,i}(2\Delta t) & a_{3,i}(2\Delta t) & a_{4,i}(2\Delta t) & a_{5,i}(2\Delta t) \\ \vdots & \vdots & \vdots & \vdots \\ a_{2,i}(T-\Delta t) & a_{3,i}(T-\Delta t) & a_{4,i}(T-\Delta t) & a_{5,i}(T-\Delta t) \end{bmatrix}, \tag{6.21}$$

where

$$a_{k,i}(t) := \frac{7!}{k!\,(7-k)!}(\kappa(t))^k\left(1-\kappa(t)\right)^{7-k}. \tag{6.22}$$

The reference of the output function y_d and the derivatives of this output function $\dot{y}_d$ and $\ddot{y}_d$ can be calculated by

$$y_{d,i}(t) := B_{s,i}(s)|_{s=\kappa(t)} \tag{6.23}$$

$$\dot{y}_{d,i}(t) := \left.\frac{\partial s}{\partial t}\frac{dB_{s,i}}{ds}\right|_{s=\kappa(t)} \tag{6.24}$$

$$\ddot{y}_{d,i}(t) := \left.\left(\frac{\partial^2 s}{\partial t^2}\frac{dB_{s,i}}{ds} + \left(\frac{\partial s}{\partial t}\right)^2 \frac{d^2 B_{s,i}}{ds^2}\right)\right|_{s=\kappa(t)}. \tag{6.25}$$

And, the output function is defined as

$$y = \begin{bmatrix} \theta_c - y_{d,\theta_c} \\ x_w - y_{d,x_w} \\ \theta_2 - y_{d,\theta_2} \\ \theta_4 - y_{d,\theta_4} \\ \phi_1 - y_{d,\phi_1} \\ \phi_2 - y_{d,\phi_2} \end{bmatrix}. \tag{6.26}$$

Using this output function, we conducted the output zeroing control as described in Section 2.2.1.

6.6 Simulation result

Using the previous control method, we show the bipedal walking simulation result in this section. For the output zeroing control gain, $K_P = 4 \times 10^4$ and $K_D = 4 \times 10^2$ were chosen where K_P is the gain of the output y, and K_D is the gain of the time derivative of output $\dot{y}$.

Figure 6.21 represents the walking gait of the bipedal robot model in one step and Figure 6.22 shows the trajectory tracking result of each joint. Table 6.5 shows the mean velocity of the gait and the cost of transport. The value of cost of transport in this simulation were observed to be higher than those observed in trajectory optimization. This result is caused by the feedback control of the states, and feedback gain affects on this value.

Table 6.5: Simulation result of bipedal robot model

	The mean velocity of walking [m/s]	Cost of Transport
value	0.4521	0.1217

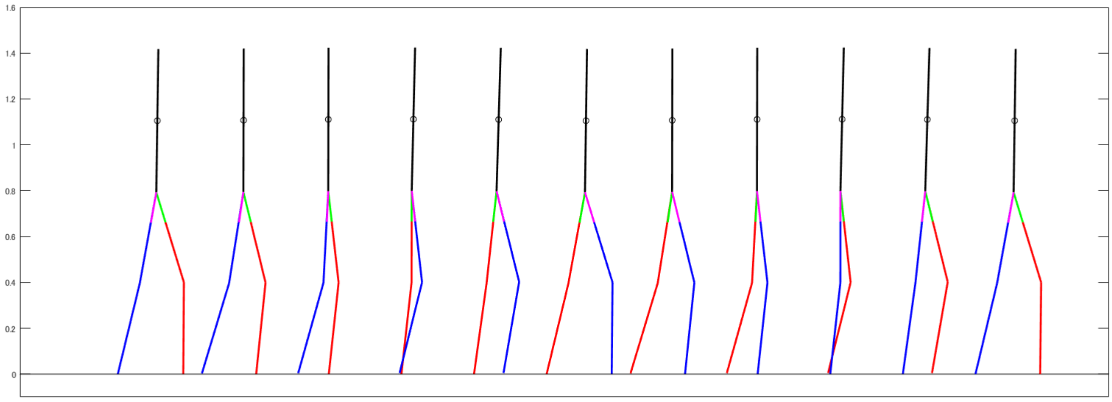


Figure 6.21: Walking simulation result of devised bipedal model

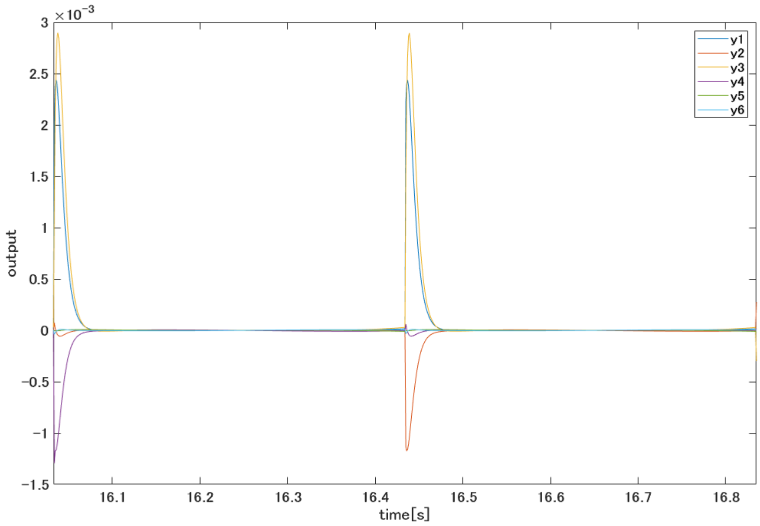


Figure 6.22: Tracking output function in one step

6.7 Conclusion

In this chapter, we verified the improvement of the performance of the walking gait using the mechanical devices for bipedal robots. We compared the Cost of Transport value as a performance of the gait, and obtained the optimized values for the parameters for the mechanical devices. The method for generating these gaits is the same method for generating the simple template model, so that the results from the complex dynamics behavior can be easily obtained. After we obtain the optimized trajectory and parameters for mechanically devised model, we also confirmed the walking simulation with feedback control using output zeroing control method.

Chapter 7

Conclusion

In this thesis, we presented control methods and mechanical devices for the bipedal robot models using a simple template model dynamics. In particular, we verified the improvement of the performance of the walking gait using this method, and we expanded this method by proposing a new template model for the walking. We also presented the method to control the mechanically devised bipedal models by using the template model, and we compared the performance of the mechanically devised systems with the nominal system by using the method which is usually used for generating the template model. The details of the contributions are described as follows:

In chapter 2, we described the derivations of some mathematical tools used in this thesis. We described the derivation of rigid body dynamics using Eular-Lagrange equation, derivation of constraint force from ground, and derivation of state changes when impact occurs, and we described some control schemes such as model matching control and output zeroing control.

In chapter 3, we presented a template model called SLIP-SL which is an extension to the popular SLIP model.The key to achieving periodic movements of this model is to modify the appropriate stiffness parameters, and we implemented this using a solver of a trajectory optimization problem. This periodic movement was generated under the condition that there is no input actuation for the template model, so that the resultant trajectory is only dependent on the natural dynamics of the simplified model. After calculating the template trajectory with appropriate parameters, we conducted the feedback control for a multi-body bipedal robot with an output zeroing method and verified the method with numerical simulation results.

In chapter 4, we presented the robustness analysis of the bipedal walking using an output zeroing control with a trajectory optimization. We conducted a numerical simulation and verified that the proposed method increases the robustness on uneven terrain compared with the previous method. Since the algorithm of optimizing trajectory is based on discretized dynamics, we used Bézier curves to interpolate the optimized discrete states. We verified the result that the proposed method improves the robustness compared with the previous research, and confirmed the stable walking with the proposed method.

In chapter 5, we presented a control method that realizes the running of bipedal robots consisting of series elastic actuators(SEA). We considered the SEA model which consists of a rotational link and a linear spring so that the conventional methods cannot be applied. And we proposed how to calculate the actuator inputs for realizing that the CoM of SEA robot models imitates the SLIP model's stable running and desired accelerations are determined by non-SEA models. We confirmed the validity of the proposed method by numerical simulation results.

In chapter 6, we present the verification of mechanical improvements for bipedal walking robots by adding series elasticity and mechanical elements to the joint structure including inerter and wobbling mass. We showed the performance improvement in walking efficiency of the improved models compared to the nominal bipedal model.

As future works for this research, the followings can be considered:

- **Expanding the idea of SLIP-SL model to the running gait**

We proposed the modified version of the SLIP model to improve the performance of the walking. The basic idea of this method is considering the swing foot dynamics in the single stance phase. In the running gait, there is an additional state phase that both feet are floating in the air. When the robot's feet are off the ground, we cannot assume the robot as a fully-actuated system any more, so a template model for considering this phase should be developed.

- **Designing a template model for the efficient 3D robot movement**

The template model used in this thesis are limited to a 2-dimensional robot's movement in sagittal plane. Since the dynamics of the 3-dimensional robot model is more complex than the 2-dimensional model, realizing the dynamical movement of the 3D robot is challenging problem in many researches. So it is expected that the template model such as SLIP model and SLIP-SL model can be modified for the 3D robot by adding the natural dynamics of 3D spring-mass structure in the coronal plane or frontal plane.

- **Output function with relative degree 3 for the robot using the SLIP-SL model**

In this thesis, we used the fully-actuated robot to track the trajectory of the SLIP-SL model. However, since the SLIP-SL model doesn't have a ankle joint, this model can be applied to the walking for the under-actuated robot. In the process of applying this template model to the under-actuated robot, it confronts the problem of the stability of the zero dynamics of the system. This problem can be solved by designing the appropriate output function with a relative degree 3, and from the result of this method, it is expected that the robustness of the gait could be improved in uneven terrain.

- **To achieve near optimal CoT by feedback control**

In chapter 6, the optimized walking was realized by the output zeroing control. However, the realized CoT was larger than the optimal value by a factor of larger than 2. So we should consider feedback control and definition of the desired walking trajectory in a continuous time domain to achieve the near optimal CoT.

Bibliography

[1] M. Kasaei, N. Lau, and A. Pereira, 2019. "A robust biped locomotion based on linear-quadratic-gaussian controller and divergent component of motion". *IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, pp. 1429–1434.

[2] T. Hayashi, and M. Yamakita, 2015. "Robust analysis of biped walking on uneven terrain using output zeroing controls". *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 505–510.

[3] F. Loeffl, A. Werner, D. Lakatos, J. Reinechke, S. Wolf, R. Burger, T. Gumpert, F. Schmidt, C. Ott, M. Grebenstein, and A. Albu-Schaffer, 2014. "The dlr c-runner: Concept, design and experiments". *Proc. of IEEE-RAS International Conference on Humanoid Robots*, pp. 2950–2957.

[4] B. H. Brown, and G. Zeglin, 1998. "The bow leg hopping robot". *Proc. of IEEE International Conference on Robotics and Automation*, pp. 781–786.

[5] P. Mann, "Chapter 1: Fully-actuated vs. Underactuated Systems", *In Lagrangian and Hamiltonian Dynamics.*, *Downloaded on 2020/11/30 from http://underactuated.mit.edu/.*

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

[7] A. Patel, S. L. Shield, S. Kazi, A. M. Johnson, and L. T. Biegler, 2019. "Contact-implicit trajectory optimization using orthogonal collocation". *IEEE Robotics and Automation Letters,* **4**(2), pp. 2242–2249.

[8] A. Hereid, M. J. Powell, and A. D. Ames, 2014. "Embedding of slip dynamics on underactuated bipedal robots through multi-objective quadratic program based control". *53rd IEEE Conference on Decision and Control*, pp. 2950–2957.

[9] J. Rummel, Y. Blum, and A. Seyfarth, 2010. "Robust and efficient walking with spring-like legs". *Bioinspiration and Biomimetics.*

[10] H. Geyer, A. Seyfarth, and R. Blickhan, 2010. "Compliant leg behaviour explains basic dynamics of walking and running". *Proceedings of the Royal Society,* **273**, pp. 2861–2867.

[11] M. A. Sharbafi, A. M. N. Rashty, C. Rode, and A. Seyfarth, 2010. "Reconstruction of human swing leg motion with passive biarticular muscle models". *Human movement science,* **273**, pp. 2861–2867.

[12] M. Kelly, 2017. "An introduction to trajectory optimization: How to do your own direct collocation". *Society for Industrial and Applied Mathematics,* **59**(4), pp. 849–904.

[13] J. Koenemann, G. Licitra, M. Alp, and M. Diehl, 2017. "Open-ocl–open optimal control library".

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

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

[16] P. A. Bhounsule, J. Cortell, A. Grewal, B. Hendriksen, J. D. Karssen, C. Paul, and A. Ruina, 2014. "Low-bandwidth reflex-based control for lower power walking: 65 km on a single battery charge". *The International Journal of Robotics Research*, pp. 1305–1321.

[17] M. Alghooneh, C. Q. Wu, and M. Esfandiari, 2016. "A passive-based physical bipedal robot with a dynamic and energy-efficient gait on the flat ground". *IEEE/ASME Transactions on Mechatronics,* **21**(4), pp. 1977–1984.

[18] T. McGeer, 1990. "Passive dynamic walking". *The International Journal of Robotics Research*, pp. 62–82.

[19] D. Hobbelen, T. d. Boer, and M. Wisse, 2008. "System overview of bipedal robots flame and tulip: Tailor-made for limit cycle walking". *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2486–2491.

[20] K. Narioka, S. Tsugawa, and K. Hosoda, 2009. "3d limit cycle walking of musculoskeletal humanoid robot with flat feet".

IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4676–4681.

[21] Y. Hanazawa, H. Suda, Y. Iemura, and M. Yamakita, 2012. "Active walking robot mimicking flat-footed passive dynamic walking". *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1281–1286.

[22] K. Sekiguchi, 2015. "Stabilization of three-link acrobot via hierarchical linearization". pp. 7808–7813.

[23] L. Cambrini, C. Chevallereau, C. H. Moog, and R. Stojic, 2000. "Stable trajectory tracking for biped robots". pp. 4815–4820.

[24] D. G. E. Hobbelen, and M. Wisse, 2007. "A disturbance rejection measure for limit cycle walkers: The gait sensitivity norm". *IEEE Transactions on Robotics,* **23**(6), pp. 1213–1224.

[25] T. Sugihara, Y. Nakamura, and H. Inoue, 2002. "Real-time humanoid motion generation through zmp manipulation based on inverted pendulum control". *Proc. of IEEE International Conference on Robotics and Automation*, pp. 1404–1409.

[26] R. Blickhan, 1989. "The spring-mass model for running and hopping". *Journal of Biomechanics*, pp. 1217–1227.

[27] M. Buhler, and D. E. Koditschek, 1988. "Analysis of a simplified hopping robot". *Proc. of IEEE International Conference on Robotics and Automation*, pp. 817–819.

[28] A. D. Ames, 2014. "Human-inspired control of bipedal walking robots". *IEEE Transactions on Automatic Control*, pp. 1115–1130.

[29] A. Giusti, J. Malzahn, N. G. Tsagarakis, and M. Althoff, 2017. "Combined inverse-dynamics/passivity-based control for robots with elastic joints". *IEEE International Conference on Robotics and Automation*, pp. 5281–5288.

[30] M. Keppler, D. Lakatos, C. Ott, and A. A. Schaffer, 2018. "Elastic structure preserving(esp) control for compliantly actuated robots". *IEEE Transactions on Robotics*(2), pp. 317–335.

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

[32] W. L. Ma, S. Kolathaya, E. R. Ambrose, C. M. Hubicki, and A. D. Ames, 2017. "Bipedal robotic running with durus-2d: Bridging the gap between theory and experiment". *Proc. of International Conference on Hybrid Systems : Computation and Control*, pp. 265–274.

[33] N. T. Doan, T. Hayashi, and M. Yamakita, 2015. "High speed running of flat foot biped robot with inerter using slip model". *Proc. of IEEE International Conference on Advanced Intelligent Mechatronics(AIM)*, pp. 110–115.

[34] J. W. Hurst, 2008. "The role and implementation of compliance in legged locomotion". *Doctoral dissertation, Robotics Institute, Carnegie Mellon Univ., Tech. Report CMU-RI-TR-08-48.*

[35] D. Lakatos, and A. Albu-Schaffer, 2016. "Modal matching: An approach to natural compliant jumping control". *IEEE ROBOTICS AND AUTOMATION LETTERS.*

[36] G. A. Pratt, and M. M. Williamson, 1995. "Series elastic actuators". *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst./Human Robot Interact. Cooperative Robots*, pp. 399–406.

[37] D. Lakatos, G. Garofalo, F. Petit, C. Ott, and A. Albu-Schaffer, 2013. "Modal limit cycle control for variable stiffness actuated robots". *2013 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4934–4941.

[38] A. Albu-Schaffer, O. Eiberger, M. Fuchs, M. Grebenstein, S. Haddadin, C. Ott, A. Stemmer, T. Wimbock, S. Wolf, C. Borst, and G. Hirzinger, 2011. "Anthropomorphic soft robotics from torque control to variable intrinsic compliance". *in Robotics Research (Springer Tracts in Advanced Robotics)*, pp. 185–207.

[39] J. Reher, E. A. Cousineau, A. Hereid, C. M. Hubicki, and A. D. Ames, 2016. "Realizing dynamic and efficient bipedal locomotion on the humanoid robot durus". *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1794–1801.

[40] M. C. Smith, 2002. "Synthesis of mechanical networks: the inerter". *IEEE Transactions on Automatic Control,* **47**(10), pp. 1648–1662.

[41] Y. Hanazawa, H. Suda, and M. Yamakita, 2011. "Analysis and experiment of flat-footed passive dynamic walker with ankle inerter". *2011 IEEE International Conference on Robotics and Biomimetics*, pp. 86–91.

[42] Y. Hanazawa, T. Hayashi, M. Yamakita, and F. Asano, 2013. "High-speed limit cycle walking for biped robots using active up-and-down motion control of wobbling mass". pp. 3649–3654.

[43] L. Rome, L. Flynn, and T. Yoo, 2006. "Biomechanics: Rubber bands reduce the cost of carrying loads". pp. 1023–1024.

[44] D. Tanaka, F. Asano, and I. Tokuda, 2012. "Gait analysis and efficiency improvement of passive dynamic walking of combined rimless wheel with wobbling mass". pp. 151–156.

[45] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. Canudas-De-Wit, and J. W. Grizzle, 2003. "Rabbit: a testbed for advanced control theory". *IEEE Control Systems Magazine,* **23**(5), pp. 57–79.