Control of a Bipedal Robot Walker on Rough Terrain

by

Hae Won Park

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Engineering)
in The University of Michigan
2012

Doctoral Committee:

Professor Jessy W. Grizzle, Chair
Professor Semyon M. Meerkov
Assistant Professor Kenn Oldham
Assistant Professor Shorya Awtar

To my parents, who made all of this possible, for their endless support and encouragement.

ACKNOWLEDGEMENTS

I would like to sincerely thank my advisor, Prof. Jessy Grizzle, for his guidance, support, patience, and most importantly, his continual inspiration throughout my study. Whenever I was in trouble with an unexpected difficulty in my research, he always inspired me with his enthusiasm and provided hints that led me to new discoveries. I would like to thank my dissertation committee members, Prof. Semyon Meerkov, Prof. Kenn Oldham, and Prof. Shorya Awtar, for their help and support. Their invaluable comments and suggestions made my thesis more complete. I also would like to thank Koushil Sreenath for being my congenial colleague, close friend, and co-investigator throughout my years in Michigan. Because of him, the thousands of hours spent in the lab to make MABEL to walk and run will always stay with me as pleasant and unforgettable memories. I am greatly indebted to Alireza Ramezani for his dedication in helping my experiments. Without his selfless assistance, most of my experimental results after Koushil's graduation would have been impossible. I am also grateful to Jonathan Hurst for designing and fabricating MABEL as well as providing me helpful advice on the design of novel mechanisms. I owe a great debt of gratitude to Gabriel Buche and Jeffrey Konscol for their contributions in the preliminary stage of my study. Their contributions were instrumental for us arriving at the successful assembly and installation of MABEL in the lab, along with fabrication of reliable and robust electrical parts. Benjamin Morris and Ioannis Poulakakis are acknowledged for their contributions to the theoretical background of my work. Lastly, I would like to thank my family and close friends, who always encouraged and inspired me. Their endless moral support was a tremendous help to me and brought this work to completion.

Hae Won Park
Ann Arbor, April 2012

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

Appendix

ABSTRACT

Control of a Bipedal Robot Walker on Rough Terrain

by

Hae Won Park

Chair: Jessy W. Grizzle

Bipedal locomotion has attracted attention for its potential ability, superior when compared to wheeled locomotion, to overcome rough terrain. Even though numerous studies have been done with the goal of realizing this ability in practice, significant restrictions still remain. Currently, bipedal robots can only accommodate unplanned obstacles that are less than 6% of leg length and ground height variations exceeding a few centimeters must be known a priori.

This dissertation is directed toward addressing these restrictions by pursuing a novel control design to allow MABEL, a new robot testbed with a compliant transmission, to accommodate unknown terrain with ground variations equaling 15% to 20% of leg length. In pursuit of this goal, first, the dynamics of MABEL are identified, and as a result, two mathematical models, one for control design and the other for controller verification, are derived. These models are used extensively in the design of a set of feedback controllers that can each accommodate a specific type of terrain variation. Next, a finite-state machine is designed that manages transitions among controllers for flat-ground walking, stepping-up, stepping-down, and a trip reflex. The design of each control mode and the transition conditions among them are conducted through a combination of constrained nonlinear optimization and heuristics based on how humans deal with obstacles. The finite-state machine

is experimentally evaluated and results in MABEL (blindly) accommodating various types of platforms, including ascent of a 12.5 cm high stair, stepping-off from an 18.5 cm high platform, and walking over a platform with multiple ascending and descending steps.

CHAPTER I

Introduction

Bipedal locomotion has attracted attention for its potential ability, superior when compared to wheeled locomotion, to overcome rough terrain or environments with discontinuous supports [88]. Thousands of years after the invention of the wheel, though wheel-based locomotion has been developed to the point that it provides fast dependable mass transportation, it lacks locomotion ability when navigating unstructured natural environments such as unpaved terrains, rocky mountains, and terrains with discontinuous supports [4, 37, 36]. Furthermore, structural environments designed to help humans move inside buildings or residences, including stairs with significant height and steep angles, and small doors sized for humans, can be challenging obstacles to wheeled locomotion because the environments are not designed for it [4]. While wheeled locomotion has difficulties in navigating these natural and man-made obstacles, humans have excellent ability to accommodate them. Despite hundreds of encounters with such obstacles in everyday life, humans rarely make mistakes leading to falls.

The W-Prize shown in Fig. 1.1 will award $ 200,000 for the first bipedal robot able to traverse the course within a specified time and a given energy budget [2, 67]. The course is composed of a size-limiting arch representing doors in a building or home, irregularly spaced discontinuous supports typical of stepping stones across a stream, and a stair case with steps of height at least 25 cm and width of less than 40 cm, which is slightly larger than a standard-sized stair case[1]. In addition, the course includes a now-you-see-it-now-you-

[1]The 1996 Council of American Building Officials (CABO) and the 2000 International Code Council recommendations call for unit runs to be not less than 10 inches and unit rises not more than 7 3/4 inches.

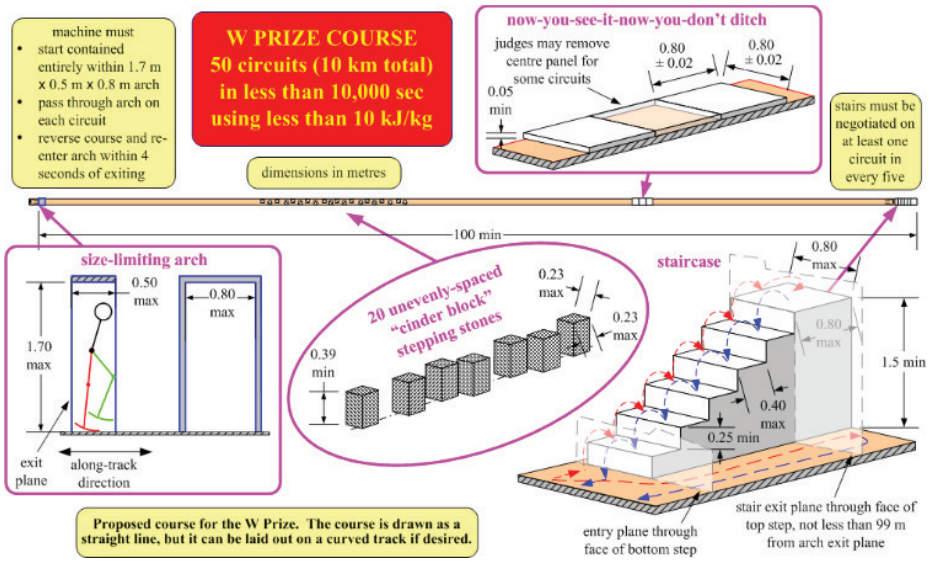


Figure 1.1: Course diagram for the W-prize. Courtesy of W-Prize organization [2].

don't ditch representing a surprise step off a curb. This obstacle is designed to require that a robot's gait not be overly dependent on vision and other external sensing for overcoming obstacles. The closed-loop system consisting of the robot and feedback control system must be robust to unplanned variations in ground height.

While important progress is being made through studies of walking over uneven ground, significant restrictions remain that prevent bipedal robot walkers from being used in practice. Existing bipedal robots can only accommodate unplanned obstacles that are less than 6% of leg length [46, 48, 121, 89, 63, 50], a value that is unrealistically small when compared to common obstacles in everyday life, such as the height of steps in a building [7, 27] or the curb height of a sidewalk on a city street. Ground height variations exceeding a few centimeters must be known a priori and require carefully planned maneuvers to overcome them. Requirements of a priori information of the ground height variations can be a significant drawback in cases when clear vision is not possible. Rescue operations in smoke-filled buildings or underground are typical examples of such cases.

This thesis is directed toward addressing the limitations of current bipedal robot research on locomotion over rough terrain by pursuing the following goal: demonstrating control solutions which allow a bipedal robot to walk over rough terrain with significant ground height variation (exceeding 20% of leg length) without any priori information or vision

feedback. The research includes fundamental theory in gait control as well as experimental verification on MABEL, a novel planar bipedal robot described in Chapter II.

Before beginning our study of walking over rough terrain, various types of rough terrain are classified and approaches to accommodate certain types of terrain are introduced.

1.1 Classification of Terrain

In real-world environments, a bipedal robot will encounter various kinds of terrain. Here, we classify terrain according to several criteria and analyze some of the difficulties in accommodating each type of terrain.

Known or Unknown Terrain: First, terrains can be categorized into known or unknown terrain depending on whether a priori information about the terrain profile is available or not. The degree of availability of terrain profile information usually yields distinguishably different controller performance for walking over rough terrain. When terrain profile information is accurately provided ahead of time, a bipedal robot can plan its walking gait in order to manage the upcoming ground variations. By changing or modifying its gait in advance, large disturbances such as stairs of considerable height can be accommodated successfully [69, 3, 56, 112, 42]. Provided with one-step ahead terrain information, a compass walker could walk drastically more steps until it falls on stochastically varying terrain than a corresponding blind walker [12].

However, in order to obtain information on terrain profiles beforehand, one or more sensors, such as a camera [69] or a range finder [56], need to be employed, and their signals need to be properly extracted to obtain correct and useful terrain profile information. Acquiring terrain profiles correctly and reliably in real time, within a reasonable sampling interval, is a very challenging task. Vision signal processing is a computationally heavy task and the quality of the processed information is highly dependent on the environment. How the gait modifications should be done as a function of the obtained information is another non trivial task to be solved. Furthermore, in extreme hazardous situations where it would be highly desirable to replace a human rescuer with a bipedal robot, such as smoke-filled environments due to fire, clear vision is not possible at all.

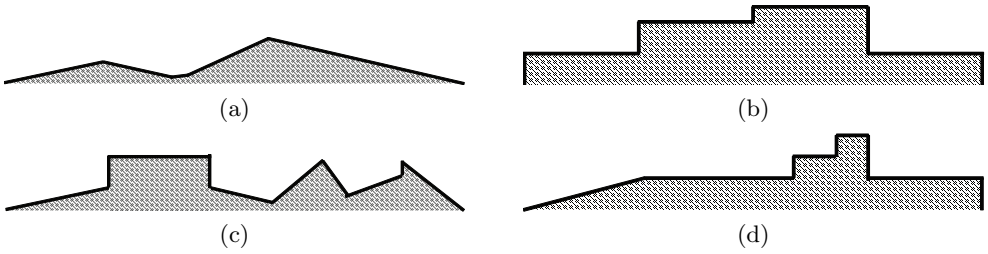


Figure 1.2: Classification of terrain into: (a) Slopes (b) Stairs (c) Randomly-varying (d) Structured.

In cases where terrain profile information is not available beforehand, bipedal robots cannot preplan to handle the obstacles. Instead, the variation in ground height should be accommodated by intrinsic robustness of the robot plus feedback controller [113, 28]. This has been approached through feedback design using robustness metrics [46, 48, 12, 11, 10], and the application of a carefully chosen reflex policy based on obtained dimension of the obstacle after the robot bumped into, or stepped off or onto it [50, 8, 16, 86, 122, 123]. The latter case utilizes calculated ground height variation using joint angles and length of the links [50, 86, 16] only, or in tandem with measured signals from a mechanism specifically designed for detection of ground variations [122, 123].

Slope or Stair: Terrain can be also classified into slopes or stairs according to its geometric shape, as shown in Figure 1.2.

To walk up or down a slope in a stable manner, the gradient of the slope has been estimated by the relative elevation between stance and swing leg for a walker with both point feet and non trivial feet [50, 86, 16, 125]. Another approach involves measuring foot inclination for a walker with non trivial feet [122, 123]. Using the measured gradient of the slope, these references apply simple adaptive strategies[2] to the walking gait, and in this manner, slopes with inclination up to 23 deg have been accommodated.

On the other hand, accommodating a stair is a more demanding task for a bipedal robot. Without a priori information about a vertical obstacle, a robot can trip over it, and eventually fall. Therefore, most of research on stair climbing is focused on the case when a priori information about the terrain is available [42, 112, 69, 3]; using a priori

[2]For example, in the research conducted in [86], desired hip height is adjusted based on simple geometric consideration in response to ground inclination.

terrain information, a robot's footholds and swing leg trajectory are carefully adjusted and controlled so that the robot avoids tripping over stair steps. A few studies have been conducted on unknown ground profiles, where a reflex strategy has been designed and employed in order to handle tripping over obstacles, and the ensuing step of the robot is planed for the safe continuation of the walking gait [8, 93].

Descending stairs can also destabilize a robot by speed increase or excessive knee bending due to large impacts with the walking surface. In this sense, a step-down test is usually considered as a robustness test for bipedal walking controller design [46, 48, 112, 89], and several approaches have been proposed to increase the maximum manageable depth of step-down. A reflexive strategy in response to a step-down disturbance has also been proposed when the robot has the ability to detect the step-down disturbance [50].

Structured or Randomly-varying Terrain: The terrain in natural environments is often complex and random (see Figure 1.2). On the other hand, human living environments, such as a building or a city street, typically consist of structured elements including flat ground with a few millimeters of height variation, vertical steps, and sloped surfaces (see Figure 1.2). While human-engineered terrain mainly consists of flat surfaces with small height variations, surface variations with considerable height, including stairs and slopes, are occasionally interspersed with flat surfaces.

Randomly-varying terrain has been modeled with a stochastic discrete-transition map [12, 57, 10] or as a discrete nonlinear system with exogenous inputs [70, 96]. An approximately-optimal controller has been designed to maximize the mean first-passage time until the biped falls [12], and a disturbance-attenuation controller has been designed to reduce the effects of the ground-variation disturbance on a particular regulated output [70, 96]. In order to make the problem mathematically tractable, simplifying assumptions are usually made, such as the robot is neither tripping nor scuffing against the ground [12, 57, 10], as well as slipping of the stance foot is not allowed. To make the problem even simpler, the nonlinear system is linearized around an operating point or a periodic orbit so that the rich theoretical framework available for linear systems can be applied [96].

When only occasional changes in ground height are encountered, a totally different framework can be employed for controller design. An encounter with a structured dis-

turbance is usually followed by a few steps on flat ground before the robot comes across another disturbance. A set of reflex-based controllers that steer the robot's perturbed state back into the basin of attraction of the controller has been designed for various kinds of disturbances [61, 50, 8, 118]. The key point is that a few steps of walking on flat ground between disturbances may be enough to return the robot's motion to a nominal gait.

1.2 Classification of Controller Design to Accommodate Rough Terrain

As noted during our classification of terrain, a very broad spectrum of ideas and approaches has been suggested in the literature to accommodate rough terrain. Here, we group various approaches that are relevant to this thesis work.

Switching Control Design: Controller design involves the development of a library of controllers as well as switching conditions managing transitions between the controllers. The switching condition can be based on the detection of a disturbance or based on detecting significant deviation of the robot's states from a nominal operating point. The library of controllers has been based on intuition or empirical knowledge [8, 50, 86, 16], or by using various optimization techniques [6, 61, 118].

Robustness Metric: Various robustness metrics have been proposed by a number of researchers [57, 46, 87, 103, 30, 29, 31, 119, 44]. Controller design is conducted by focusing on the improvement of the proposed robustness metric. Several approaches based on parameter sweeps [48, 47] or dynamic programming [10] have been suggested to improve the robustness metric.

Analytical Methods: Controller is designed by the use of a common set of mathematical tools. Some of the commonly used methods are aperiodic gait design by a sequence of virtual constraints obtained using hybrid zero dynamics [124], passivity-based control and controlled symmetries [38, 97], stabilization of transversal dynamics [63], and linear matrix inequalities [96].

Machine Learning or Dynamic Programming: Control actions are tuned by a machine learning algorithm [71, 45, 109, 113, 92, 90] or dynamic programming [118, 101, 70] to yield increased rewards or cost function value. This control design is usually computa-

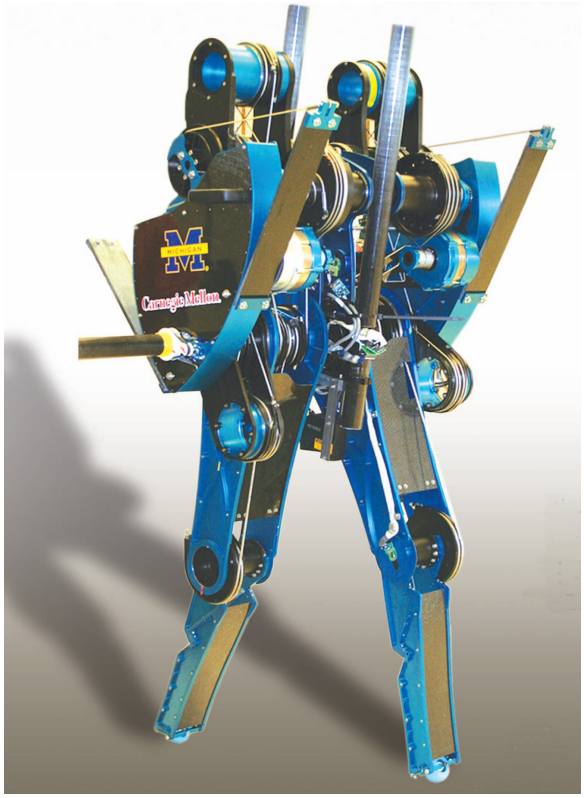


Figure 1.3: MABEL, an experimental testbed for bipedal locomotion. Courtesy of IEEE Control Systems Magazine and Catharine June (EECS Dept.).

tionally costly, and not suitable for unstable systems or systems with many states.

Biomimetic Control Design: A controller is designed based on mechanics and principles of human walking. Motion capture data of humans climbing up and down stairs, and uneven ground are used to design virtual constraints [83, 78]. A control design that mimics human muscle reflexes is proposed in [33], and yields stable walking on sloped surfaces.

1.3 Control Design for Walking over Rough Terrain with MABEL

This thesis work is aimed toward control design that allows MABEL, shown in Figure 1.3, to accommodate unknown structured terrain with obstacles consisting of slopes and vertical steps. In pursuit of this goal, the dynamics of MABEL are identified in the first part of the thesis. Two mathematical models, one for control design and the other for controller verification, are carefully derived, and their unknown parameters are identified through modular and sequential identification procedures. The first model, termed

the "simplified design model", captures MABEL's natural compliance from its designed-in physical springs as well as its drivetrain dynamics. The second model, termed the "detailed model", additionally captures unexpected compliance due to cable stretch in the transmission mechanism as well as a model of the laboratory surface on which MABEL walks. In the second part of the thesis, an extension of the framework of switching control that was initiated in [124] is explored to design a controller which allows MABEL to accommodate unexpected large ground variations without prior knowledge of ground profiles. The control design focuses on accommodating various types of obstacles including step-down, step-up, and tripping while utilizing MABEL's compliance and natural dynamics. The simplified design model is extensively used to design the controller through an iterative optimization process. Modifications to the controller are subsequently made by hand account for discrepancies between the simplified design model and the detailed model. The designed controller's performance is demonstrated through a set of experimental results conducted on various types of ground profiles.

1.4 Organizations of Dissertation

Chapter II overviews the planar bipedal robot MABEL, with emphasis on its mechanical structure, actuation, and sensing. The key feature in MABEL's mechanism is the use of a cable-based differential transmission designed to realize two novel features: unique actuated coordinates; and a large spring in series with DC motors actuating the knee joints. The reasoning behind the two novel features is addressed, and how these features are realized by the cable differential mechanism is explained. Two mathematical models for the robot are derived and identified in Chapters III and IV. The simplified design model is a six degree of freedom hybrid model consisting of a continuous stance phase, an instantaneous double support phase, and a switching surface that manages transitions between them. Based on the assumption of a pinned connection between the stance toe and the ground, zero stretch in the cables, and no deflection in the spring of the swing leg, the continuous stance dynamics is derived using Lagrange's method. The instantaneous double support phase is derived by modeling the impact of the swing toe with the ground as an inelastic

contact between two rigid bodies. The transition map which takes pre-impact states to post-impact states is obtained using the principles of virtual work and conservation of generalized momentum. A detailed model that is appropriate for controller verification is derived next. By representing cable stretch as a spring-damper, incorporating the boom dynamics to account for asymmetry side-to-side, and representing the contact between robot and ground as being comprised of compliant elements and a LuGre friction model [13] [80], the detailed model is obtained and the accuracy of the model is significantly improved over that of the simplified design model. These models will be used extensively throughout the dissertation.

Chapters V, VI, and VII provide control designs which allow MABEL to accommodate terrains with significant variations in heights. First, the step-down problem is addressed in Chapter V. Preliminary experimental work on the step-down test [73] revealed that a baseline feedback controller designed for walking on flat ground overreacts when correcting the forward pitching motion of the torso, causing a second, very rapid, forward pitching motion of the torso. Because the angle of the swing leg was controlled relative to the torso, the swing leg rotated forward rapidly as well and impacted the ground with sufficient force to break the leg. A new controller is designed to attenuate torso oscillation after the step-down event. This controller is triggered when the calculated depth of the step-down exceeds a threshold value, replacing the baseline controller for one step. A dramatic increase in performance is obtained: MABEL stepping off 20.32 cm platform without falling.

Chapter VI explains the design of controllers for accommodating step-up and tripping events. A novel switch mechanism is designed and installed on the front of the robot's shin to detect impacts with vertical obstacles. Based on the on-off signal from the designed switch along with the robot's state measured by joint encoders, a detected impact is categorized into early or late tripping, and a corresponding controller is activated to handle the event. In Chapter VII, the set of reflexive controllers designed in Chapters V and VI is combined in a finite-state machine. The finite-state machine leads to impressive experimental results including blindly walking over obstacles with 12.5 cm increase in height and 18.5 cm decrease in depth. Limitations of the controller and future directions of research are given in Chapter VIII.

Appendix A presents novel mechanisms which have been designed and installed on MABEL as part of this thesis work. Appendices B, C, and D explain how motor-torque bias is estimated, how pulley inertias are lumped, and how stable hopping is achieved in the process of the system identification. Modification on virtual compliance to account for dynamic asymmetry of side-to-side due to motion constraints posed by the boom are provided in Appendix E. Appendix F presents modification of the swing leg virtual constraints for elevating swing leg rapidly, which is activated in response to early tripping over obstacle. Appendix G displays sequential screen captures taken from the videos of the experiments conducted in Chapters VI and VII. Appendix H addresses an outer-loop discrete-event-based control design which focuses on attenuating influences of randomly-varying ground variations on the output to be regulated.

CHAPTER II

Mechanism Overview

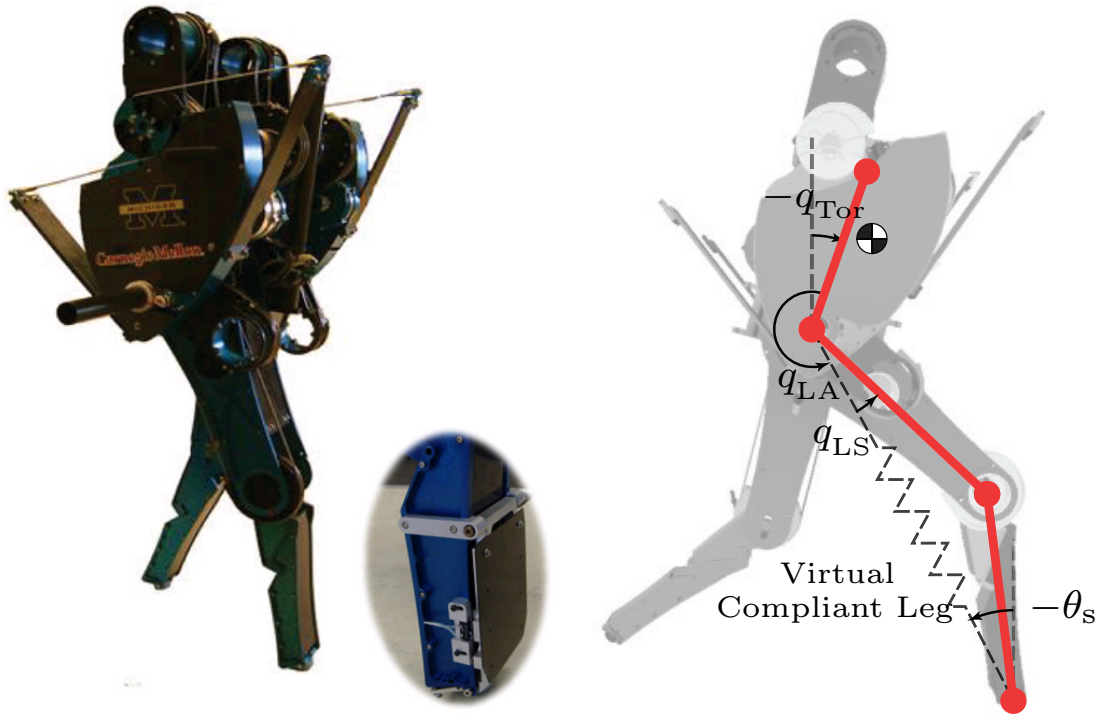
In this chapter, we briefly introduce MABEL, the platform used to test feedback controllers developed in this dissertation.

MABEL's body consists of five links, namely, a torso, two thighs, and two shins. As shown in Figure 2.1, the legs are terminated in point feet. All actuators are located in the torso, so that the legs are kept as light as possible[1]. The hip of the robot is attached to a boom, as shown in Figure 2.2. The robot's motion is therefore tangent to a sphere centered at the pivot point of the boom on the central tower. The boom is 2.25 m long, and while the resulting walking motion of the robot is circular, it approximates the motion of a planar biped walking along a straight line.

Unlike most bipedal robots, the actuated degrees of freedom of each leg do not correspond to the knee and the hip angles as in a conventional biped. Instead, as shown in figures 2.3 and 2.5, for each leg, a collection of differentials is used to connect two motors to the hip and knee joints in such a way that one motor controls the angle of the virtual leg defined by the line connecting the hip to the toe, while the second motor is connected in series with a spring to control the length of the virtual leg. Conventional bipedal robot coordinates and MABEL's actuated coordinates, which are depicted in Figure 2.5, are related by

$$q_{\mathrm{LA}} = \frac{1}{2}\left(q_{\mathrm{Th}} + q_{\mathrm{Sh}}\right) \tag{2.1}$$

[1]Light legs enable rapid swing leg motion. Rapid swing leg motion is required for fast gaits [100], and for rapid-elevation of the swing leg, which in Chapter VI, is activated in response to tripping over obstacles.



(a) (b)

Figure 2.1: MABEL's general morphology and powertrain. (a) MABEL is shown with the switch in front of the shin that is used to detect contact with obstacles. The robot is planar, with a boom providing stabilization in the frontal plane. The robot weighs 65 kg and is 1 m at the hip. (b) The *virtual compliant leg* created by the drivetrain through a set of differentials.

and

$$q_{\mathrm{LS}} = \frac{1}{2}\left(q_{\mathrm{Th}} - q_{\mathrm{Sh}}\right), \tag{2.2}$$

where LA stands for leg angle, LS stands for leg shape, Th stands for thigh, and Sh stands for shin.

The springs in MABEL isolate the leg-shape motors from the impact forces at leg touchdown. In addition, the springs store energy during the compression phase of a running gait, when the support leg must decelerate the downward motion of the robot's center of mass. The energy stored in the spring can then be used to redirect the center of mass upwards for the subsequent flight phase, propelling the robot off the ground. As explained in [54, 41, 53], both of these properties, shock isolation and energy storage, enhance the energy efficiency of running and reduce the actuator power requirements. Similar advantages are also present

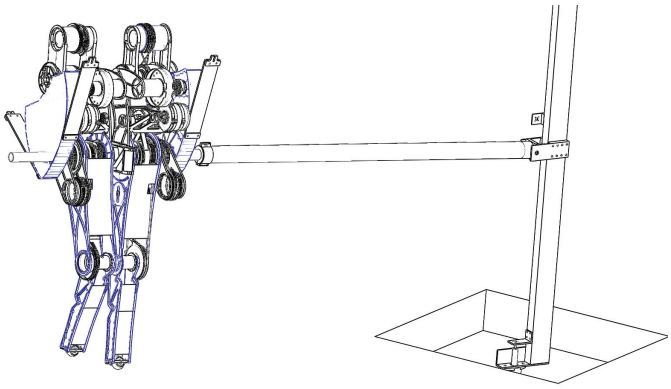
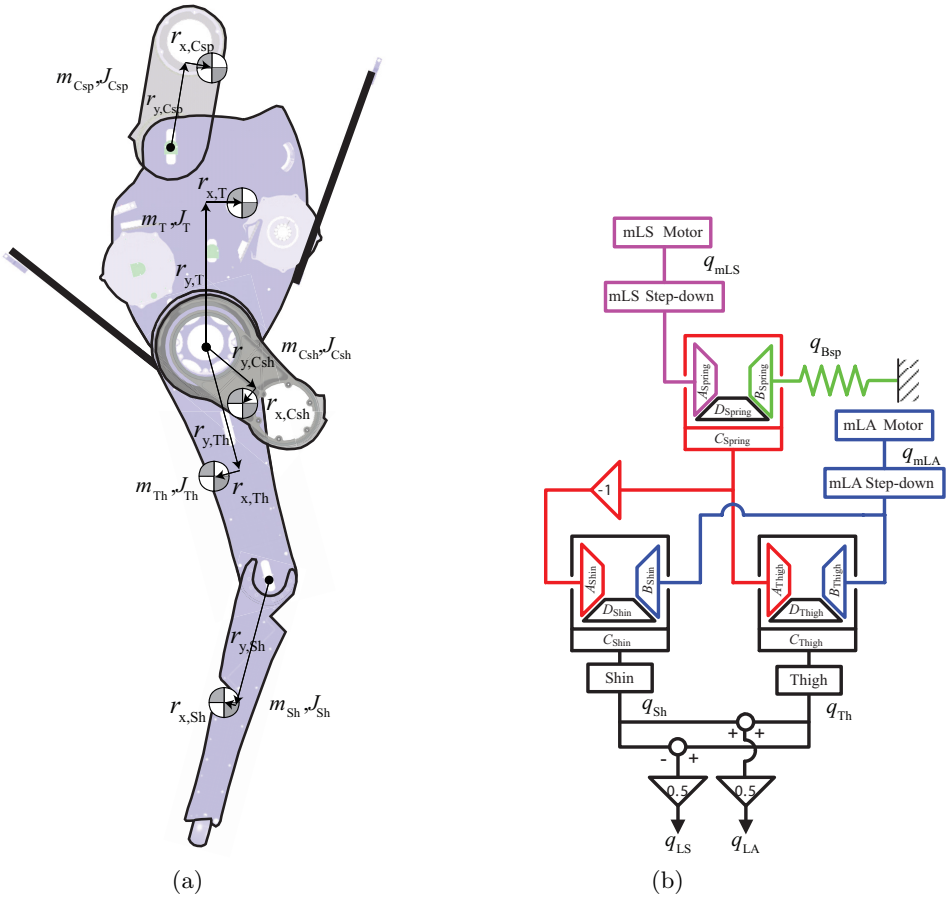


Figure 2.2: Approximate planar motion. The boom constrains MABEL's motion to the surface of a sphere centered at the attachment point of the boom to the central tower. The boom is approximately 2.25 m long. The central tower is supported on a slip ring through which power and digital communication lines, such as E-stop and ethernet, are passed.



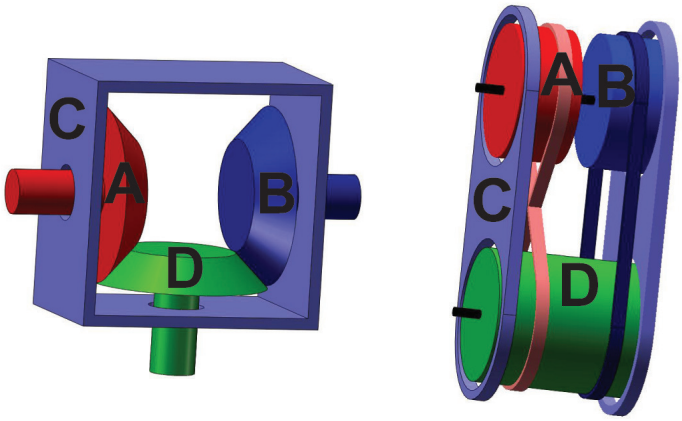
(a) (b)

Figure 2.3: Robot and transmission mechanism. (a) Links comprising MABEL. Csp, T, Csh, Th, and Sh denote C_{Spring}, Torso, C_{Shin}, Thigh, and Shin, respectively. (b) The transmission mechanism consists of spring, thigh, and shin differentials. The spring differential realizes a serial connection between the leg-shape motor and the spring. The thigh differential moves the thigh link in the leg, while the shin differential moves the shin link. The gear ratios are described in figures 2.6 and 2.7.

in walking on flat ground, but to a lesser extent compared to running on uneven terrain, due to the lower forces at leg impact and the reduced vertical travel of the center of mass. The robotics literature strongly suggests that shock isolation and compliance are useful for walking on uneven terrain [19, 20, 49, 43, 72, 91, 106].

2.1 Transmission

The transmission mechanism for each half of MABEL consists of three cable differentials, labeled the spring, thigh, and shin differentials, and a spring, as shown in Figure 2.3(b). The thigh and shin differentials translate shin angle and thigh angle into leg shape and leg angle. Thus, the electric motors control the leg angle and leg shape. The spring differential forms a series connection between a spring and the motor for leg shape so that the resulting system behaves approximately like a pogo stick. In order to keep the legs light, the motors and



(a) (b)

Figure 2.4: Two versions of a differential mechanism. (a) shows a conventional gear differential, while (b) shows a cable differential. Each differential consists of three components, labeled A, B, and C, connected by an internal, unobserved, idler D. In (a), A, B, and D are gears, while in (b), they are pulleys. In both (a) and (b), C is a link. The kinematic equations for a differential are given by $(q_A + q_B)/2 = q_C$ and $(q_A - q_B)/2 = q_D$, assuming equal gear ratios, where q_A, q_B, q_C, and q_D denote the angular displacements of the components.

differentials are mounted in the torso. Instead of the gear differentials depicted in Figure 2.4, cable differentials are used to achieve low friction and backlash. Although cable differentials and gear differentials have different assemblies, they work in the same manner. Each consists

of three components, labeled A, B, and C, connected by an internal, unobserved, idler D. When the gear ratios are all equal, the components A and B are constrained such that the average motion of the two is equal to the motion of the component C. Consequently, A and B can move in opposite directions if C is held stationary, and the motion of C is half of A if B is held stationary. In other words, $(q_A + q_B)/2 = q_C$ and $(q_A - q_B)/2 = q_D$, where q_A, q_B, q_C, and q_D denote the angular displacements of the components.

In MABEL's transmission mechanism, A and B are inputs to the differential, while C is an output. In the following, A_{Shin}, B_{Shin} and C_{Shin} refer to the components A, B, and C of the shin differential and likewise for the remaining two differentials. C_{Thigh} and C_{Shin} in Figure 2.3(b) are attached to the thigh and shin links, respectively. The pulleys B_{Thigh} and B_{Shin} are both connected to the leg-angle motor. The pulleys A_{Thigh} and A_{Shin} are connected to the pulley C_{Spring}, which is the output pulley of the spring differential. The spring on each side of the robot is implemented with two fiberglass plates connected in parallel to the differentials through cables, as shown in Figure 2.1. Due to the cables, the springs are unilateral, meaning they can compress but not extend; this aspect is discussed below.

Figures 2.6 and 2.7 illustrate how this transmission works when one of the coordinates q_{LA} or q_{LS} is actuated, while the other coordinate is held fixed. The path from spring displacement to rotation in q_{LS} is similar. The net motion in q_{LS} from the leg-shape motor and the spring is the sum of the individual motions.

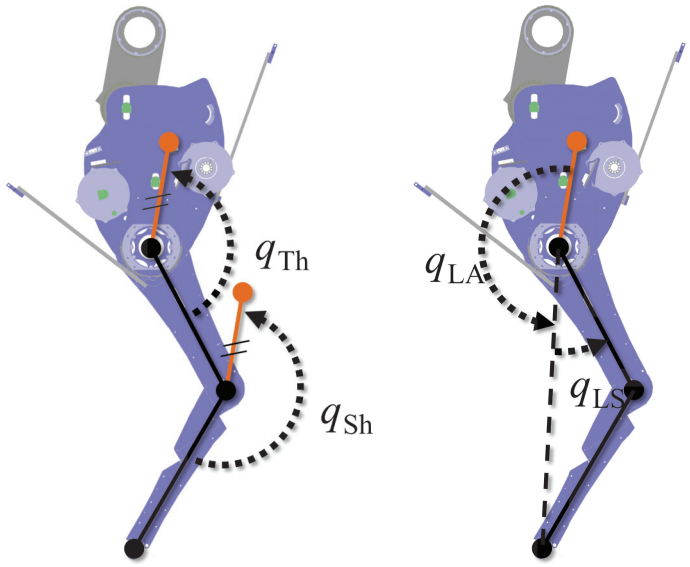
If we assume that the cables do not stretch, then the leg angles q_{LA} and q_{LS} are related to the corresponding motor angles and the angle of the pulley B_{Spring} by

$$q_{\text{LA}} = \frac{1}{\gamma_{\text{LA}\to\text{mLA}}} q_{\text{mLA}} \tag{2.3}$$

and

$$q_{\text{LS}} = \frac{1}{\gamma_{\text{LS}\to\text{mLS}}} q_{\text{mLS}} + \frac{1}{\gamma_{\text{LS}\to\text{Bsp}}} q_{\text{Bsp}}, \tag{2.4}$$

where $\gamma_{\text{LS}\to\text{mLS}} = 31.42$, $\gamma_{\text{LA}\to\text{mLA}} = -23.53$, and $\gamma_{\text{LS}\to\text{Bsp}} = 5.18$ are the gear ratios from LS to mLS, from LA to mLA, and from LS to Bsp. The calculated angles q_{LS} and q_{LA} are also



(a) (b)

Figure 2.5: Actuated coordinates. (a) shows a more conventional choice of actuated coordinates. The actuated coordinates on MABEL, shown in (b), correspond to controlling the length and orientation of the virtual prismatic leg indicated by the dashed line connecting the toe to the hip. The counterclockwise direction is positive.

logged during the experiments.

2.2 Variable names

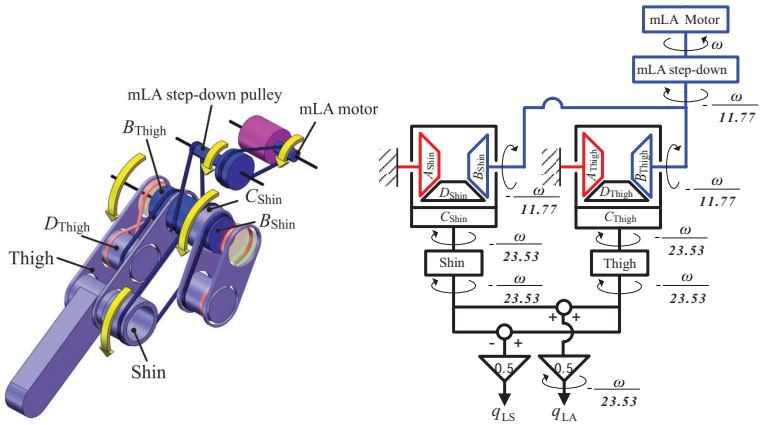
To name the variables appearing in the robot, we use the index set

$$\mathcal{I} = \{\mathrm{mLS_L},\ \mathrm{mLA_L},\ \mathrm{mLS_R},\ \mathrm{mLA_R}\}, \tag{2.5}$$

where the subscripts L and R mean left and right, mLS means motor leg shape, and mLA means motor leg angle; see Figure 2.3(b). For the links, we define the index set

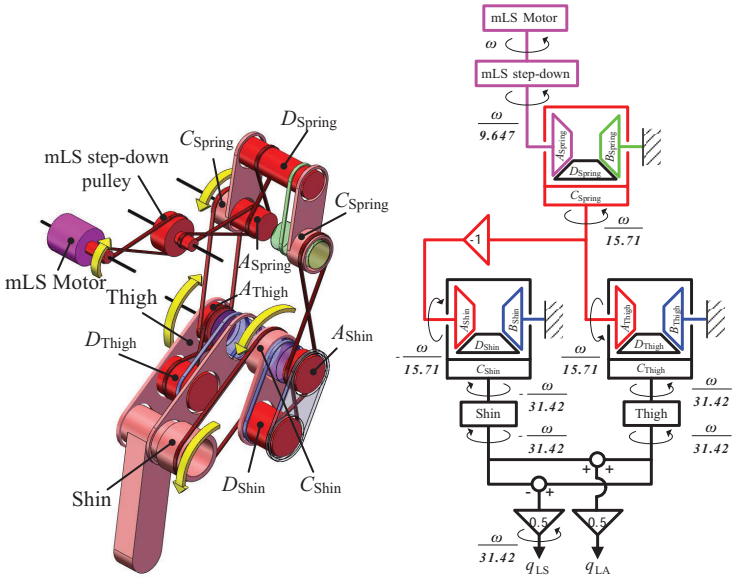
$$\mathcal{L} = \{\mathrm{T}, \mathrm{Csp}, \mathrm{Th}, \mathrm{Sh}, \mathrm{Csh}, \mathrm{Boom}\}, \tag{2.6}$$

where T, Csp, Th, Sh, Csh, and Boom represent Torso, C_{Spring}, Thigh, Shin, C_{Shin}, and Boom, respectively, as depicted in Figure 2.3(a). For the transmission mechanism, we define



(a) q_{LA} pulley system (b) q_{LA} Transmission

Figure 2.6: Leg-angle actuation. (a) Torque from the leg-angle motor is transmitted to q_{LA} as defined in Figure 2.5 through one step-down pulley and two differentials, namely, Thigh and Shin. (b) uses gear differentials to depict the actuation of q_{LA}, with the indicated gear ratios.



(a) q_{LS} pulley system (b) q_{LS} transmission

Figure 2.7: Leg-shape actuation. (a) Torque from the leg-shape motor is transmitted to q_{LS} as defined in Figure 2.5 through one step-down pulley and three differentials, namely, Spring, Thigh, and Shin. (b) uses gear differentials to depict the actuation of q_{LS}, with the indicated gear ratios.

the index set

$$\mathcal{T} = \{\text{Asp}, \text{Bsp}, \text{Dsp}, \text{Ath}, \text{Bth}, \text{Dth}, \text{Ash}, \\ \text{Bsh}, \text{Dsh}, \text{mLSsd}, \text{mLAsd}, \text{mLS}, \text{mLA}\}, \tag{2.7}$$

where A, B, and D correspond to the components of the differentials in Figure 2.3(b), and sp, th, sh, and sd denote spring, thigh, shin, and step down, respectively, as depicted in Figure 2.3. Throughout this article, the notation for coordinates and torques in Table 2.1 is used.

Table 2.1: Notation for MABEL's coordinates and torques. The subscripts L and R denote the left leg and right leg, respectively.

$q_{\text{LS}_{\text{L,R}}}$	leg-shape angle
$q_{\text{mLS}_{\text{L,R}}}$	motor leg-shape angle
$q_{\text{LA}_{\text{L,R}}}$	leg-angle angle
$q_{\text{mLA}_{\text{L,R}}}$	motor leg-angle angle
$q_{\text{Bsp}_{\text{L,R}}}$	pulley-B_{Spring} angle
$\tau_{\text{mLS}_{\text{L,R}}}$	leg-shape motor torque
$\tau_{\text{mLA}_{\text{L,R}}}$	leg-angle motor torque
$\tau_{\text{Bsp}_{\text{L,R}}}$	pulley-B_{Spring} torque

2.3 Sensors

MABEL is equipped with encoders and contact switches, but not force sensors[2]. On each side of the robot, magnetic encoders for measuring joint angles are present on the pulleys mLA, mLS, Dth, Bsp, and mLAsd, as well as the knee joint. From the geometry of the legs, the leg-shape angle q_{LS} is one half of the knee joint angle. Each encoder has a resolution of 2048 counts per revolution, and thus 0.1758 deg/count.

The angle of the thigh with respect to the torso is measured by an encoder that has 2048 counts per revolution and a 19.8:1 gear ratio, for a total of 40,550.4 counts per revolution, and thus 0.008878 deg/count. The angle of the torso with respect to the vertical is measured by an encoder that has 2048 counts per revolution and a 3:1 gear ratio, for a total of 6144

[2]The feedback control designs developed in this thesis do not utilize any type of force measurement. Position and velocity of joint angles are utilized to calculate control torque, and contact switches are used to detect swing leg impacts with the ground. Force sensors were initially used on Rabbit [14], but were not able to withstand the impulsive forces of ground impacts. On MABEL, the springs in the leg-shape motors can be used to estimate the ground reaction force in the direction of the virtual leg.

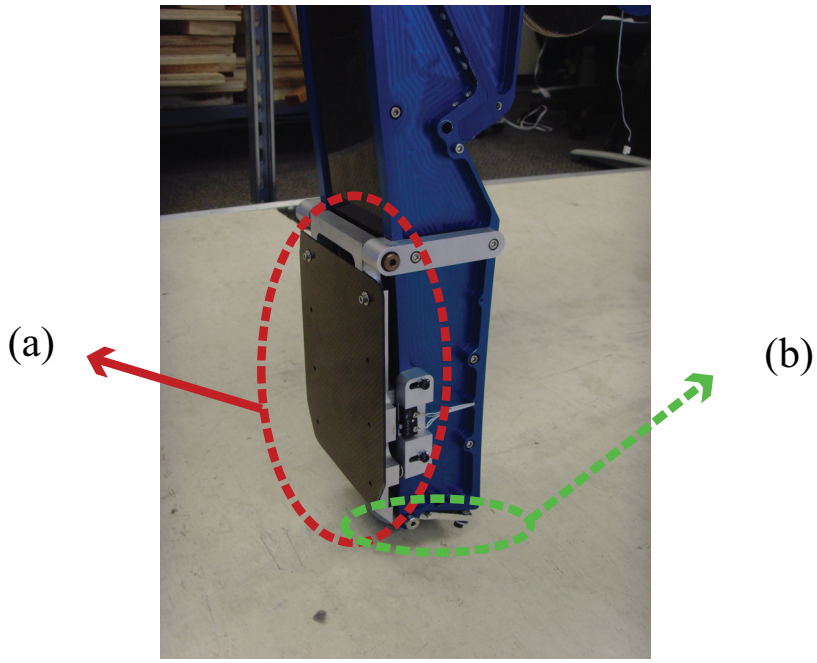


Figure 2.8: Switches installed on MABEL. (a) Contact switch on the front on the shin, (b) Contact switch on the bottom of the toe.

counts per revolution, and thus 0.05859 deg/count. The pitch angle of the boom is measured by an encoder with 2048 counts per revolution and a 17.6:1 gear ratio, for a total of 36,044.8 counts per revolution, and thus 0.09988 deg/count. The rotation angle of the central tower is measured by an encoder with 40,000 counts per revolution and a 7.3987:1 gear ratio, for a total of 295,948 counts per revolution, and thus 0.001216 deg/count.

As depicted in Figure 2.8, MABEL is equipped with contact switches installed on the bottom of the toe and the front of the shin. The contact switches installed on the bottom of the toe are used for detecting impacts with the ground, whereas the contact switches installed on the front of the shin are used for detecting contact with vertical obstacles.

Additional contact switches are installed at the hard stops on various pulleys to detect excessive rotation within MABEL's workspace. If a contact switch on a hard stop closes, power to the robot is immediately turned off by the computer.

Because MABEL does not have velocity sensors, angular velocities must be estimated by numerically differentiating position signals. For real-time applications, such as feedback control, causal methods for numerical differentiation are required [18, 66]. For offline applications, such as parameter identification, acausal or smoothing algorithms can be employed

[105]. We use the spline interpolation method of [23], which can be used in a causal or acausal manner.

2.4 Embedded Computer and Data Acquisition

The onboard computer is a 1.3-GHz Intel Celeron M CPU running a QNX real-time operating system. The RHexLib library, originally developed for the robot RHex [91], is used for implementing control-algorithm modules as well as for logging data over the network data and communicating with the robot. A user interface for monitoring the robot's state on a secondary Linux-based system uses utilities provided by RHexLib.

Digital and analog IO are handled with compactPCI data-acquisition cards from Acromag. A custom-built compactPCI module houses the interface circuitry between the Acromag data-acquisition cards and the sensors on the robot.

The sample period of the embedded controller is set at 1 ms. Though a 1.5 ms sample period is probably adequate, the power to the robot is automatically shut off by a watchdog timer if any controller update cycle exceeds 1 ms. Data acquisition and logging consume approximately 0.5 ms of each sample period, leaving 0.5 ms for control computations and signal processing. PD-based controllers can be implemented in the available processing time without special programming considerations. For controllers based on inverse dynamics, such as those reported in [100], the real-time calculations are performed in approximately 0.11 ms with a public-domain C++-template library from Boost [1], which provides standard matrix-algebra operations.

CHAPTER III

Mathematical Model

Two models of MABEL have been developed and identified in [100, 77]. This section briefly summarizes a *simplified design model that is appropriate for control design* and a more *detailed model appropriate for controller verification*.

These two models will be used extensively in Chapters V, VI, and VII. The simplified design model will be used for controller design via iterative optimization because it can be simulated twenty times faster than the detailed model. The detailed model will be used to verify controller performance.

3.1 Simplified Design Model

The hybrid model consists of a continuous-time stance phase and an instantaneous double support phase. The overall dynamic model is derived with the method of Lagrange [59].

3.1.1 Stance Phase

For modeling the stance phase, the stance toe is assumed to act as a passive pivot joint (no slip, no rebound and no actuation). The springs in the transmission are appropriately chosen to support the entire weight of the robot, and hence are stiff. Consequently, it is assumed that the spring on the swing leg does not deflect, that is, $q_{\mathrm{Bsp}_{\mathrm{sw}}} \equiv 0$.

With these assumptions, the generalized coordinates are taken as,

$$q_{\mathrm{s}} := (q_{\mathrm{LA}_{\mathrm{st}}}; q_{\mathrm{mLS}_{\mathrm{st}}}; q_{\mathrm{Bsp}_{\mathrm{st}}}; q_{\mathrm{LA}_{\mathrm{sw}}}; q_{\mathrm{mLS}_{\mathrm{sw}}}; q_{\mathrm{T}}) \in \mathcal{Q}_{\mathrm{s}}, \tag{3.1}$$

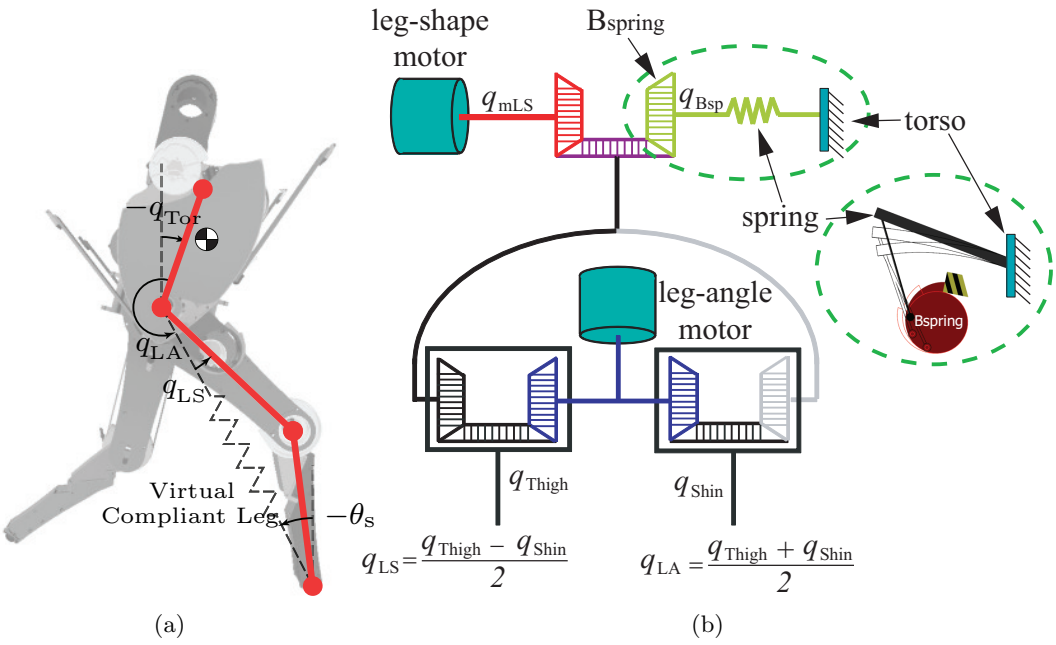


Figure 3.1: The virtual Compliant leg and powertrain of MABEL. (a) The *virtual compliant leg* created by the drivetrain through a set of differentials. (b) MABEL's powertrain (same for each leg), all housed in the torso. Two motors and a spring are connected to the traditional hip and knee joints via three differentials, which are connected such that the actuated variables are leg angle and leg shape, see Figure 3.1a, and so that the spring is in series with the leg-shape motor. The differentials are realized with pulleys and cables.

where $\mathcal{Q}_{\text{s}}$ is a simply-connected, subset of $\mathbb{S}^6$, and as in Figure 3.1a and Figure 3.1b, $q_{\text{LA}_{\text{st}}}, q_{\text{mLS}_{\text{st}}}$, and $q_{\text{Bsp}_{\text{st}}}$ are the leg angle, leg-shape motor position, and angle of the pulley B_{spring} (a pulley which is connected to the free end of the spring as shown in Figure 3.1b, and therefore corresponds to spring deflection) of the stance leg, respectively; $q_{\text{LA}_{\text{sw}}}$ and $q_{\text{mLS}_{\text{sw}}}$ are the leg angle and leg-shape motor position of the swing leg, respectively; q_{T} is the angle of torso with respect to the vertical.

The Lagrangian for the stance dynamics, $\mathcal{L}_{\text{s}} : TQ_{\text{s}} \to \mathbb{R}$ is defined by

$$\mathcal{L}_{\text{s}} = \mathcal{K}_{\text{s}} - \mathcal{V}_{\text{s}}, \tag{3.2}$$

where, $\mathcal{K}_{\text{s}} : TQ_{\text{s}} \to \mathbb{R}$, and $\mathcal{V}_{\text{s}} : Q_{\text{s}} \to \mathbb{R}$ are the total kinetic and potential energies of the mechanism respectively. The total kinetic energy is obtained by summing the kinetic energy of the linkage, $\mathcal{K}_{\text{s}}^{link}$, the kinetic energy of the stance and swing leg transmissions,

$\mathcal{K}_{\mathrm{s}}^{trans_{\mathrm{st}}}, \mathcal{K}_{\mathrm{s}}^{trans_{\mathrm{sw}}}$, and the kinetic energy of the boom, $\mathcal{K}_{\mathrm{s}}^{boom}$.

$$\mathcal{K}_{\mathrm{s}}\left(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}}\right) = \mathcal{K}_{\mathrm{s}}^{link}\left(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}}\right) + \mathcal{K}_{\mathrm{s}}^{trans_{\mathrm{st}}}\left(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}}\right) + \mathcal{K}_{\mathrm{s}}^{trans_{\mathrm{sw}}}\left(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}}\right) + \mathcal{K}_{\mathrm{s}}^{boom}\left(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}}\right). \tag{3.3}$$

The linkage model is standard; the boom is modeled as in [114, p. 94]. Physically, the boom constraints the robot to move on the surface of a sphere, and a full 3D model would be required to accurately model the robot and boom system. However, we assume the motion to be planar, and only consider the effects due to mass and inertia of the boom. This will introduce some discrepancies between simulation and experimental results. The symbolic expressions for the transmission model are available online at [40].

Similar notation is used for the potential energy,

$$\mathcal{V}_{\mathrm{s}}\left(q_{\mathrm{s}}\right) = \mathcal{V}_{\mathrm{s}}^{link}\left(q_{\mathrm{s}}\right) + \mathcal{V}_{\mathrm{s}}^{trans_{\mathrm{st}}}\left(q_{\mathrm{s}}\right) + \mathcal{V}_{\mathrm{s}}^{trans_{\mathrm{sw}}}\left(q_{\mathrm{s}}\right) + \mathcal{V}_{\mathrm{s}}^{boom}. \tag{3.4}$$

Due to its unilateral nature, the spring is not included in the potential energy of the transmission; only the mass of the motors and pulleys is included. The unilateral spring is considered as an external input to the system.

With the above considerations, the robot dynamics for stance phase can be determined through Lagrange's equations

$$\frac{d}{dt}\frac{\partial \mathcal{L}_{\mathrm{s}}}{\partial \dot{q}_{\mathrm{s}}} - \frac{\partial \mathcal{L}_{\mathrm{s}}}{\partial q_{\mathrm{s}}} = \Gamma_{\mathrm{s}}, \tag{3.5}$$

where, Γ_{s} is the vector of generalized forces acting on the robot, and can be written as,

$$\Gamma_{\mathrm{s}} = B_{\mathrm{s}} u + E_{\mathrm{ext}}\left(q_{\mathrm{s}}\right) F_{\mathrm{ext}} + B_{fric}\tau_{fric}\left(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}}\right) + B_{sp}\tau_{sp}\left(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}}\right), \tag{3.6}$$

where the matrices B_{s}, E_{ext}, B_{fric}, and B_{sp} are derived from the principle of virtual work and define how the actuator torques u, the external forces F_{ext} at the leg, the joint friction forces τ_{fric}, and the spring torques τ_{sp} enter the model respectively.

Applying Lagrange's equations (3.5), with the kinetic and potential energies defined by

(3.3), (3.4), respectively, results in the second-order dynamical model

$$D_{\mathrm{s}}\left(q_{\mathrm{s}}\right)\ddot{q}_{\mathrm{s}}+C_{\mathrm{s}}\left(q_{\mathrm{s}},\dot{q}_{\mathrm{s}}\right)\dot{q}_{\mathrm{s}}+G_{\mathrm{s}}\left(q_{\mathrm{s}}\right)=\Gamma_{\mathrm{s}} \tag{3.7}$$

for the stance dynamics of MABEL. Here, D_{s} is the inertia matrix, the matrix C_{s} contains Coriolis and centrifugal terms, and G_{s} is the gravity vector.

The state-variable form of the stance-phase dynamics, with state vector $x_{\mathrm{s}} := (q_{\mathrm{s}}; \dot{q}_{\mathrm{s}}) \in T\mathcal{Q}_s$, can be expressed as,

$$\dot{x}_{\mathrm{s}} := \begin{bmatrix} \dot{q}_{\mathrm{s}} \\ -D^{-1}M \end{bmatrix} + \begin{bmatrix} 0 \\ D^{-1}B \end{bmatrix} \tag{3.8}$$

$$:= f(x_{\mathrm{s}}) + g(x_{\mathrm{s}})u, \tag{3.9}$$

where, f and g are the drift and input vector fields, and $M := C(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}})\dot{q}_{\mathrm{s}} + G(q_{\mathrm{s}}) - B_{fric}\tau_{fric}(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}}) - B_{sp}\tau_{sp}(q_{\mathrm{s}}, \dot{q}_{\mathrm{s}})$.

3.1.2 Double Support Phase

An impact occurs when the swing leg touches the ground, modeled here as an inelastic contact between two rigid bodies. It is assumed that there is neither rebound nor slip at impact and the post-transition spring position on the new swing leg is non-deflected. In addition to modeling the impact of the leg with the ground and the associated discontinuity in the generalized velocities of the robot [51], the transition map accounts for the assumption that the spring on the swing leg is at its rest length, and for the relabeling of robot's coordinates so that only one stance model is necessary [100].

Mathematically, the impact occurs when the solution of (3.8) intersects the switching surface

$$\mathcal{S} := \left\{ x_{\mathrm{s}} \in T\mathcal{Q}_s \,\middle|\, p^v_{\mathrm{toe_{sw}}}(q_{\mathrm{s}}) = 0 \right\}, \tag{3.10}$$

where $p^v_{\mathrm{toe_{sw}}}(q_{\mathrm{s}})$ is vertical position of the swing toe.

When impact occurs, the method of [52] provides a (static) map that takes the state

variables just before impact to their values just after impact,

$$x_{\mathrm{s}}^{+} = \Delta(x_{\mathrm{s}}^{-}). \tag{3.11}$$

3.1.3 Modification for Ground Variations

In this thesis work, the switching surface in (3.10), is modified to account for a change in ground height at impact,

$$\mathcal{S}^{H} := \left\{ x_{\mathrm{s}} \in T\mathcal{Q}_{\mathrm{s}} \,\middle|\, p^{v}_{\mathrm{toe_{sw}}}(q_{\mathrm{s}}) = H, H \in \mathbb{R} \right\}, \tag{3.12}$$

where H is the height of the platform. With this definition, the original switching surface with a platform height of zero is denoted by $\mathcal{S}^{0}$.

3.1.4 Hybrid Model of Walking

Together, the stance-phase dynamics (3.8) and reset map (3.11) form a nonlinear system with impulse effects

$$\begin{cases} \dot{x}_{\mathrm{s}} &= f(x_{\mathrm{s}}) + g(x_{\mathrm{s}})u \quad x_{\mathrm{s}} \notin \mathcal{S}^{H} \\ x_{\mathrm{s}}^{+} &= \Delta(x_{\mathrm{s}}^{-}) \qquad\qquad\quad x_{\mathrm{s}} \in \mathcal{S}^{H}. \end{cases} \tag{3.13}$$

More details about the development of the impact map and the hybrid model for MABEL are presented in [100].

3.2 Detailed Model

The simplified design model does not fully reflect experimental reality due to the following reasons: cable stretch in the robot's drivetrain; asymmetry due to the boom radius not being large enough; the fact that the simplified impact model assumes an instantaneous double support phase, whereas, in experiments, the double support phase lasts approximately 20 ms. More details are provided in [41], [100] and [77]. By representing cable stretch as a spring-damper, and incorporating the boom dynamics to account for asymmetry side-to-side and a ground model comprised of compliant ground and LuGre friction

model [13, 80] into the mathematical model, the accuracy of the model is significantly improved. However, because of the complexity of this model, simulations of the *detailed model* take 20 times longer than the *simplified design model*. Hence, this model is not appropriate for optimization processes which may require thousands of simulations.

Therefore, to take advantage of each model's strengths, which are low computational effort for the simplified design model and high accuracy for the detailed model, iterative controller design is conducted on the simplified design model, and then the designed controller is tested on the detailed model, before implementing it on the robot.

The overall dynamic model is derived with the method of Lagrange [59]. When computing the Lagrangian, it is convenient to consider the spring torques, the cable-stretch torques, the ground reaction forces and the joint-friction torques as external inputs to the model. The generalized coordinates of the detailed model are taken as

$$q_{\mathrm{d}} := (q_{\mathrm{LA}_L}; q_{\mathrm{mLA}_L}; q_{\mathrm{LS}_L}; q_{\mathrm{mLS}_L}; q_{\mathrm{Bsp}_L}; q_{\mathrm{LA}_R}; q_{\mathrm{mLA}_R}; q_{\mathrm{LS}_R}; q_{\mathrm{mLS}_R}; q_{\mathrm{Bsp}_R}; q_{\mathrm{Tor}}; p^h_{\mathrm{hip}}; p^v_{\mathrm{hip}}) \in \mathcal{Q}_{\mathrm{h}}, \tag{3.14}$$

where $\mathcal{Q}_{\mathrm{d}}$ is a simply-connected, subset of $\mathbb{S}^{11} \times \mathbb{R}^2$, and, as in Figure 3.1a and Figure 3.1b, q_{LA}, q_{mLA}, q_{LS}, and q_{mLS} are the leg angle, leg-angle motor position, leg shape, and leg-shape motor position, respectively, q_{Tor} is the angle of the torso with respect to the vertical, and p^h_{hip} and p^v_{hip} are the horizontal and vertical positions of the hip in the sagittal plane. Subsubscript L and R denote the left and right leg, respectively. This choice of the generalized coordinates is also in agreement with the set of measured encoder signals available on MABEL and therefore, real hardware can be easily substituted for the detailed model once controller validation is completed. The model is can be expressed in second-order form as

$$D_{\mathrm{d}}(q_{\mathrm{d}})\,\ddot{q}_{\mathrm{d}} + C_{\mathrm{d}}(q_{\mathrm{d}}, \dot{q}_{\mathrm{d}})\,\dot{q}_{\mathrm{d}} + G_{\mathrm{d}}(q_{\mathrm{d}}) = \Gamma_{\mathrm{d}}, \tag{3.15}$$

where the vector of generalized forces and torques Γ_{d} acting on the robot is given by

$$\begin{aligned}\Gamma_{\mathrm{d}} = B_{\mathrm{d}}u + B_{\mathrm{fric}}\tau_{\mathrm{fric}}\left(q_{\mathrm{d}}, \dot{q}_{\mathrm{d}}\right) + B_{\mathrm{sp}}\tau_{\mathrm{Bsp}}\left(q_{\mathrm{d}}, \dot{q}_{\mathrm{d}}\right) + \\ \frac{\partial p_{\mathrm{toe}}}{\partial q_{\mathrm{d}}}^{T} F + B_{\mathrm{cable}}\tau_{\mathrm{cable}}\left(q_{\mathrm{d}}, \dot{q}_{\mathrm{d}}\right).\end{aligned} \tag{3.16}$$

Here, p_{toe} is the position vector of the two leg ends, F is the ground reaction forces on the two legs, and the matrices B_{d}, B_{fric}, B_{sp}, and B_{cable}, which are derived from the principle of virtual work, define how the actuator torques τ, the joint friction torques τ_{fric}, the spring torques τ_{Bsp}, and the cable-stretch torques τ_{cable} enter the model, respectively.

In the detailed model, the robot's state with relabeling with stance and swing can be obtained by,

$$\begin{cases} \hat{x}_{\mathrm{d}} = \hat{x}_{\mathrm{d}} & \hat{x}_{\mathrm{d}} \notin \mathcal{S}_{\mathrm{d}}^{H} \\ \hat{x}_{\mathrm{d}} = \Delta_{\mathrm{d}}(\hat{x}_{\mathrm{d}}) & \hat{x}_{\mathrm{d}} \in \mathcal{S}_{\mathrm{d}}^{H} \end{cases}, \tag{3.17}$$

where Δ_{d} is the relabeling matrix for detailed model[1], and $\mathcal{S}_{H}^{\mathrm{d}}$ is the switching surface which is given by,

$$\mathcal{S}_{\mathrm{d}}^{H} := \left\{ \hat{x}_{\mathrm{d}} \in T\mathcal{Q}_{\mathrm{d}} \left| p^{v}_{\mathrm{toe_{sw},d}}(\hat{x}_{\mathrm{d}}) = H, H \in \mathbb{R}, s \geq s_{\mathrm{early}} \right. \right\}, \tag{3.18}$$

where $p^{v}_{\mathrm{toe_{sw},d}}(\hat{x}_{\mathrm{d}})$ is the vertical position of the swing toe, and s is a monotonically increasing scalar-valued function of the gait, taking values in $[0, 1]$ (a formal definition is given in (5.4)) and represents, roughly speaking, the relative position of the robot with respect to the desired end of the step. s_{early} is the threshold value for the declaration of impact and the swapping of stance and swing legs which follows the declaration of the impact. The minimum value of s for allowing the declaration of impact is set to avoid premature swapping of the legs.

[1]The derivation of the relabeling matrix is trivial and not addressed in this thesis.

3.2.1 Nonlinear Spring

The model for the unilateral spring is augmented with terms to represent the hard stop, yielding

$$\tau_{\mathrm{Bsp}} = \begin{cases} -K_{\mathrm{B}} q_{\mathrm{Bsp}} - K d_{\mathrm{B}} \dot{q}_{\mathrm{Bsp}}, & q_{\mathrm{Bsp}} > 0, \\ -K_{\mathrm{B}} q_{\mathrm{Bsp}} - K_{\mathrm{d1}} q_{\mathrm{Bsp}}^3 - K_{\mathrm{vd1}} \dot{q}_{\mathrm{Bsp}}, & q_{\mathrm{Bsp}} \leq 0 \text{ and } \dot{\mathrm{q}}_{\mathrm{Bsp}} \geq 0, \\ -K_{\mathrm{B}} q_{\mathrm{Bsp}} - K_{\mathrm{d1}} q_{\mathrm{Bsp}}^3 - K_{\mathrm{vd1}} \dot{q}_{\mathrm{Bsp}} & \\ \quad - K_{\mathrm{vd2}} \sqrt{|\dot{q}_{\mathrm{Bsp}}|} \mathrm{sgn}(\dot{q}_{\mathrm{Bsp}}), & q_{\mathrm{Bsp}} \leq 0 \text{ and } \dot{\mathrm{q}}_{\mathrm{Bsp}} < 0, \end{cases} \tag{3.19}$$

where K_{B} and Kd_{B} corresponds to the spring constants and damping constants, respectively, and where the remaining parameters, K_{vd1}, K_{d1}, and K_{vd2} corresponds to damping constant, and nonlinear spring and damping constants of hard stop. When $q_{\mathrm{Bsp}} > 0$, the spring is deflected and the model is a linear spring-damper. When $q_{\mathrm{Bsp}} \leq 0$, the pulley is against the hard stop, a stiff damper. This model captures the unilateral nature of MABEL's built-in compliance.

3.2.2 Cable Stretch

Experiments reported in [100] show that the cables used in the differentials of MABEL stretch a significant amount under loads typical of bipedal walking. This compliance breaks the relations in (2.3) and (2.4). Consequently, q_{LA} and q_{mLA} are independent degrees of freedom, as are q_{LS}, q_{mLS}, and q_{Bsp}.

We take into account the stretching of the cables with a spring-damper model. First, the relations (2.3) and (2.4) are expressed in the form of the constraint

$$c(q) = \begin{bmatrix} q_{\mathrm{mLA}} - \gamma_{\mathrm{LA}\to\mathrm{mLA}} q_{\mathrm{LA}} \\ q_{\mathrm{mLS}} - \gamma_{\mathrm{LS}\to\mathrm{mLS}} q_{\mathrm{LS}} + (\gamma_{\mathrm{LS}\to\mathrm{mLS}} / \gamma_{\mathrm{LS}\to\mathrm{Bsp}}) q_{\mathrm{Bsp}} \end{bmatrix}, \tag{3.20}$$

where q is the vector of generalized coordinates for the robot dynamics and $c(q) \equiv 0$ corresponds to no cable stretch. Because the cable-stretch torques act on these constraints, the principle of virtual work yields the input matrix B_{cable} for the cable-stretch torques,

given by

$$B_{\text{cable}} = \frac{\partial c}{\partial q}^T . \tag{3.21}$$

Note that we are representing the forces generated in the cables as equivalent torques acting in series with the pulleys in the differentials. We furthermore assume that the cable-stretch torques can be modeled as a linear spring-damper. Therefore, for each side of the robot, the torque from the cable stretch is modeled as

$$\tau_{\text{cable}}\,(q, \dot{q}) = K_{\text{C}}\; c\,(q) + Kd_{\text{C}}\; c\,(\dot{q})\,, \tag{3.22}$$

where K_{C} is a 2×2 diagonal matrix of spring coefficients, and Kd_{C} is a 2×2 diagonal matrix of damping coefficients. The spring and damping coefficients of the cables are identified in the next section.

3.2.3 Compliant Ground

The ground reaction forces at the leg ends are based on the compliant ground model in [65, 13], using the modifications presented in [80].

Figure 3.2 illustrates robot's leg penetrating the ground, where p_{toe} and v_{toe} are the position and velocity of foot respectively. From the p_{toe} and v_{toe} along with terrain profile, the penetrating depth z_{G}, penetrating velocity $\dot{z}_{\text{G}}$, and relative velocity of the contacting surfaces v can be calculated by trivial geometry analysis. Then, the normal force F_{n} and tangential force F_{t} acting on the end of each leg are determined by

$$F_{\text{n}} = -\lambda_{\text{v}}^{a}|z_{\text{G}}|^{n}\dot{z}_{\text{G}} - \lambda_{\text{v}}^{b}|z_{\text{G}}|^{n}\text{sgn}(\dot{z}_{\text{G}})\sqrt{|\dot{z}_{\text{G}}|} + k|z_{\text{G}}|^{n}, \tag{3.23}$$

$$F_{\text{t}} = \mu(d, v)|F_{\text{n}}|, \tag{3.24}$$

where

$$\dot{d} = v - |v|\frac{\sigma_{\text{h0}}}{\alpha_{\text{h0}}}d, \tag{3.25}$$

$$\mu(d, v) = \sigma_{\text{h0}}d + \sigma_{\text{h1}}\dot{d} + \alpha_{\text{h2}}v, \tag{3.26}$$

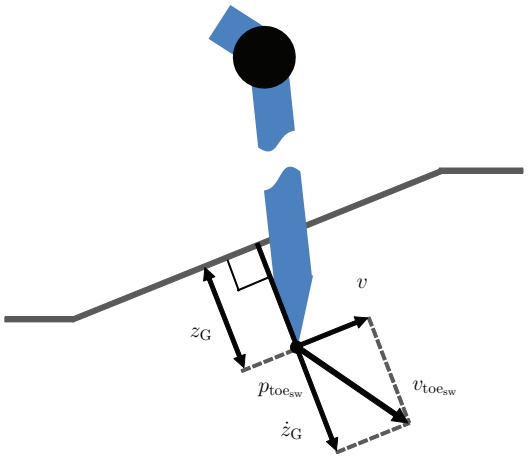


Figure 3.2: Illustration of robot's leg penetrating the ground. p_{toe} and v_{toe} are the position and velocity of swing foot respectively, v is the relative velocity of the contacting surfaces, and z_{G} and $\dot{z}_{\mathrm{G}}$ are the penetration depth and its velocity.

when the relative penetration depth z_{G} of a leg into the ground is positive, and are zero otherwise. The normal force F_{n} corresponds to a nonlinear spring-damper, with damping coefficient λ_{v}^{a} and spring stiffness k_{v}. According to [65], the exponent n, which depends on the shapes of the surfaces that are in contact, equals 1.5 when the leg end is spherical, which is roughly the case for MABEL. The signed-square-root term on penetration velocity, with coefficient λ_{v}^{b}, induces finite-time convergence of the ground penetration depth when the robot is standing still.

The tangential force F_{t} is in the form of a friction model with variable coefficient of friction μ determined by the LuGre model [13]. The LuGre model represents the interface between the two contacting surfaces as bristles, modeled by springs and dampers, which, if the applied tangential force is sufficient, are deflecting and slipping. The average deflection d of the bristles is the internal state of the friction model, v is the relative velocity of the contacting surfaces, σ_{h1} is the damping coefficient, and α_{h2} is the coefficient of viscous friction. The stiffness of the spring is σ_{h0} and the coefficient of static friction is α_{h0}.

3.3 Change of Coordinates Between Two Models

In order to verify a controller designed on the basis of the simplified design model by simulating it on the detailed model, the initial condition of the detailed model simulation

should be obtained from the initial condition of the simulation data from the simplified design model . Assume that the detailed model simulation starts with left leg being the stance leg and the right leg being the swing leg. The subscripts L and R in (3.14) can then be replaced with st and sw, as shown in (3.1). The coordinates of the detailed model are then given by

$$\hat{q}_{\mathrm{d}} := (q_{\mathrm{LA_{st}}}; q_{\mathrm{mLA_{st}}}; q_{\mathrm{LS_{st}}}; q_{\mathrm{mLS_{st}}}; q_{\mathrm{Bsp_{st}}}; q_{\mathrm{LA_{sw}}}; q_{\mathrm{mLA_{sw}}}; q_{\mathrm{LS_{sw}}}; q_{\mathrm{mLS_{sw}}}; q_{\mathrm{Bsp_{sw}}}; q_{\mathrm{Tor}}; p^h_{\mathrm{hip}}; p^v_{\mathrm{hip}}). \tag{3.27}$$

Because the states $q_{\mathrm{mLA_{st,sw}}}$, $q_{\mathrm{LS_{st,sw}}}$, $p^{h,v}_{\mathrm{hip}}$, and $q_{\mathrm{Bsp_{sw}}}$ are in (3.27), but not in (3.1), these states should be reconstructed from the states in (3.1). By the same token, the associated velocity states also need to be reconstructed. The reconstruction of the absent states is done as follows:

$$\begin{aligned}
q_{\mathrm{mLA_{st,sw}}} &= \gamma_{\mathrm{LA}\to\mathrm{mLA}} q_{\mathrm{LA_{st,sw}}} \\
q_{\mathrm{Bsp_{sw}}} &= 0 \\
q_{\mathrm{LS_{st,sw}}} &= \frac{1}{\gamma_{\mathrm{LS}\to\mathrm{mLS}}} q_{\mathrm{mLS_{st,sw}}} + \frac{1}{\gamma_{\mathrm{LS}\to\mathrm{Bsp}}} q_{\mathrm{Bsp_{st,sw}}} \\
p^h_{\mathrm{hip}} &= f^h_{\mathrm{hip}}(q_{\mathrm{s}}) \\
p^v_{\mathrm{hip}} &= f^v_{\mathrm{hip}}(q_{\mathrm{s}}) \\
\dot{q}_{\mathrm{mLA_{st,sw}}} &= \gamma_{\mathrm{LA}\to\mathrm{mLA}} \dot{q}_{\mathrm{LA_{st,sw}}} \\
\dot{q}_{\mathrm{Bsp_{sw}}} &= 0 \\
\dot{q}_{\mathrm{LS_{st,sw}}} &= \frac{1}{\gamma_{\mathrm{LS}\to\mathrm{mLS}}} \dot{q}_{\mathrm{mLS_{st,sw}}} + \frac{1}{\gamma_{\mathrm{LS}\to\mathrm{Bsp}}} \dot{q}_{\mathrm{Bsp_{st,sw}}} \\
\dot{p}^h_{\mathrm{hip}} &= \frac{\partial f^h_{\mathrm{hip}}}{\partial q_{\mathrm{s}}} \dot{q}_{\mathrm{s}} \\
\dot{p}^v_{\mathrm{hip}} &= \frac{\partial f^v_{\mathrm{hip}}}{\partial q_{\mathrm{s}}} \dot{q}_{\mathrm{s}},
\end{aligned} \tag{3.28}$$

where f^h_{hip} and f^v_{hip} are functions calculating the horizontal and vertical positions of the hip.

On the other hand, only the states in (3.1) and their associated velocities will be used in

control implementation because controller is being designed on the simplified design model. Therefore, states which are in (3.27), but not in (3.1) will not be used and are neglected in the control design. Although those absent states are neglected in the calculation of the control input, the robustness of the proposed controller will compensate for the neglected states, and, furthermore, the designed controller will be verified on the detailed model before experimental deployment.

CHAPTER IV

Parameter Identification

The primary objective of this chapter is to identify the parameters that appear in the dynamic models of MABEL that were introduced in Chapter III. The parameters we seek to identify include inertias, center-of-mass locations, spring constants, motor torque constants, friction coefficients, and power-amplifier biases.

4.1 Literature Survey

The problem of parameter identification for robot models is well studied in the literature [59, 105, 5, 32]. Most results are based on the analysis of the input-output behavior of the robot during a planned motion, where the parameter values are obtained by minimizing the difference between a function of the measured robot variables and the output of the model. An illustration of this approach is presented in [105] for identifying parameters in industrial manipulators. The standard rigid-body model is rewritten in the parametric form $\tau = \phi(q, \dot{q}, \ddot{q})\theta$, which is linear in the unknown parameters, where q, $\dot{q}$, $\ddot{q}$ are the position, velocities, and accelerations of the joints, τ is the vector of joint torques, θ is the unknown parameter vector, and ϕ is the regressor matrix. Optimization is used to define trajectories that enhance the condition number of ϕ, and these trajectories are then executed on the robot. Weighted least-squares estimation is applied to estimate the parameters, which in turn are validated by torque prediction. This approach requires acceleration, which must be estimated numerically from measured position.

An alternative approach explored in [5] uses force- and torque-sensor measurements to

avoid the need to estimate acceleration. The robot model is represented in Newton-Euler form, and a six-element wrench at the robot's wrist is expressed in a form that is linear in the unknown parameters. Force and torque at the wrist are measured directly through force and torque sensors, and parameter estimation is accomplished from this data without the need for acceleration measurements. Another class of methods [32] uses an energy-based model that uses velocity and position variables, but does not require acceleration. This method, however, relies on the integration of the input torques and the joint velocities to compute energy, which is problematic if the torque estimates are corrupted by an unknown bias.

4.2 Challenges

Parameter identification for MABEL is a challenging task for several reasons. First, MABEL has position encoders at the motors and joints, as well as contact switches at the leg ends, but lacks force and torque sensors. We thus use commanded motor torques as inputs and motor and joint position encoders as outputs in order to extract model parameters. Due to the quantization error of the encoders, it is difficult to estimate acceleration by numerically differentiating encoder signals. Hence, we estimate parameters without calculating acceleration from position data. Second, the actuator characteristics are poorly known. The motors used in MABEL are brushless direct current (BLDC) motors, which are custom manufactured on demand. Due to the small production numbers, the rotor inertias and torque constants may differ by 20% from the values supplied by the manufacturer. Figure 4.1 shows typical joint-excitation experimental data and predicted data from simulation with the motor torque constant and rotor inertias supplied by the manufacturer. We can observe that magnitudes of the two signals are clearly different. After obtaining the rotor inertias and torque constants from the identification process, simulation was able to predict experimental data with better accuracy as shown in Figure 4.7. These parameters must therefore be included in the identification procedure.

In combination with power amplifiers from a second manufacturer, the motors exhibit some directional bias. Complicating matters further, this bias varies among individual

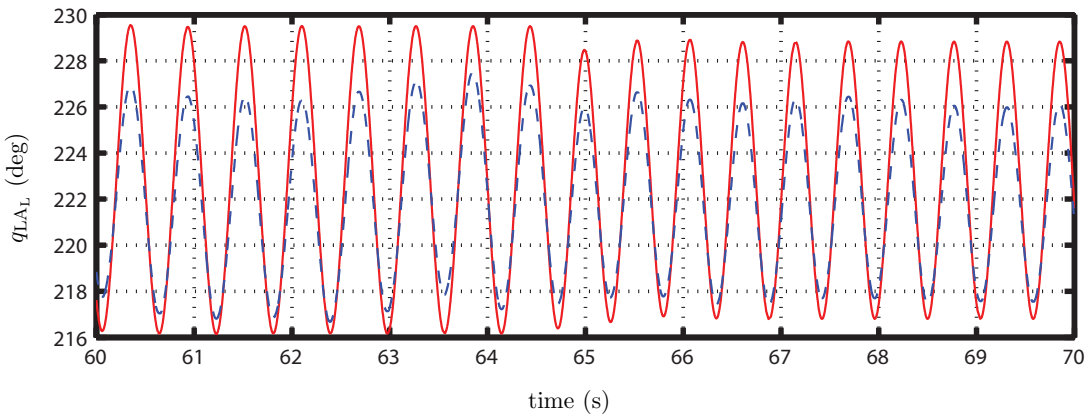


Figure 4.1: Experimental data and simulation data with the motor torque constant and rotor inertias supplied by the manufacturer. The joint positions for the simulation are indicated by solid red lines, while the joint positions for the experiments are indicated by dotted blue lines.

amplifier-motor pairs. Consequently, the amplifier bias must be considered in the identification process. A third issue affecting parameter identification is that the choice of exciting trajectory is restricted due to limitations of MABEL's work space. For example, a constant-velocity experiment for estimating friction coefficients is not feasible because the maximum range of rotation of each joint is less than 180 deg. Finally, because MABEL has many degrees of freedom, actuating all of them at once would require estimating 62 parameters simultaneously. For this reason, we take advantage of the modular nature of the robot to design experiments that allow us to sequentially build the model element by element, estimating only a few parameters at each stage of the process.

4.3 Parameter Identification Procedure

CAD packages provide estimates of the masses of the links and pulleys comprising MABEL, their lengths and radii, centers of mass, and moments of inertia. If we also account for the location and mass distribution of items not normally represented in a CAD drawing, such as ball bearing shape and density, the length and density of pulley cables, electrical wiring, onboard power electronics, and actuators and sensors, then the mechanical parameters of the robot can be estimated. Consequently, one goal of the identification procedure is to validate these estimates by comparing predicted responses to experimental

Table 4.1: Parameters to be identified, where i $\in \mathcal{I}$, $\ell \in \mathcal{L}$, and t $\in \mathcal{T}$. The subscripts L and R denote the left leg and right leg, respectively.

Differentials and Motors	
K_{i}	motor torque constant
$J_{\mathrm{rotor,i}}$	inertia of the rotor
J_{t}	inertia of the transmission pulleys
μ_{i}	friction coefficient
b_{i}	motor bias
Thigh and Shin (Leg)	
m_ℓ	mass of the link ℓ
J_ℓ	inertia of the link ℓ
$m_\ell r_{\mathrm{x},\ell}$	center of mass in x-direction of the link ℓ multiplied by mass of the link ℓ
$m_\ell r_{\mathrm{y},\ell}$	center of mass in y-direction of the link ℓ multiplied by mass of the link ℓ
Compliance (Spring)	
$K_{\mathrm{B_{L,R}}}$	spring stiffness
$Kd_{\mathrm{B_{L,R}}}$	spring damping coefficient
$K_{\mathrm{C,i}}$	cable stretch stiffness
$Kd_{\mathrm{C,i}}$	cable stretch damping coefficient

data.

In addition, model parameters for which reliable estimates are not available from CAD drawings include motor torque constants, rotor inertias, spring stiffness, and pre-load. Finally, friction parameters cannot be estimated by a CAD program and must be determined experimentally. The parameters to be identified are shown in Table 4.1.

4.3.1 Steps in the Identification Process

The first phase of the identification process focuses on estimating the actuator parameters and the friction parameters in the transmission, as well as validating the pulley inertia estimates provided by the CAD program. The torque constant K_{T} and rotor inertia J_{rotor} of each motor are also determined by analyzing the motors in series with a chain of known inertias formed by selectively coupling the pulleys that form the three differentials, as shown in Figures 2.6 and 2.7. Because the pulleys are connected by low-stretch steel cables to form a one-degree-of-freedom system, various paths in the transmission mechanism can be modeled by lumping the moments of inertia of the pulleys. This lumped moment of inertia can

be calculated by the CAD model and added to the rotor inertia of the motor. In addition, this lumped moment of inertia can be obtained from experiments. These data can be used to estimate motor torque constants, rotor inertias, viscous friction, and motor torque biases.

Next, the legs are coupled to the transmission to validate the actuation-transmission model in conjunction with the center of mass and moments of inertias of the links constituting the thigh and shin, as provided by the CAD model. For these experiments, the compliance is removed from the system by locking the orientation of the pulley B_{Spring}; the torso is held in a fixed position as well. Following this experiment, the torso's inertial parameters are estimated for validation.

Compliances are determined last. Two sources of compliance are present in the robot. One source is the unilateral, fiberglass spring designed into the transmission. The other source, which is unplanned, arises from stretching of the cables between the pulleys. The compliance of the unilateral spring is obtained from static experiments, and the compliance due to cable stretch is estimated from dynamic experiments that apply high torques to the robot.

4.3.2 Experimental Setup for the Motor, Differential, and Leg Parameters

The first phase of the identification process uses the setup depicted in Figure 4.2. The torso is fixed relative to the world frame, and the legs can freely move. The position of the pulley B_{Spring} is fixed as well, removing compliance for the initial identification phase. Torque commands are recorded and sent to the amplifiers. In turn, the amplifiers regulate the currents in the motor windings, thereby setting the motor torque values. The rotation of the motors is transmitted to the thigh and shin links through the transmission differentials as shown in Figures 2.6, 2.7, and 4.2.

The leg angles q_{LA} and q_{LS} are related to the corresponding motor angles and the angle of the pulley B_{Spring} by (2.3) and (2.4), and the calculated angles q_{LS} and q_{LA} are also logged during the experiments.

The relations (2.3) and (2.4) hold under the assumption that the cables do not stretch, which is the approximation made here, because no external loads are applied to the legs or the pulleys. When the robot is walking, the transmission system is heavily loaded due

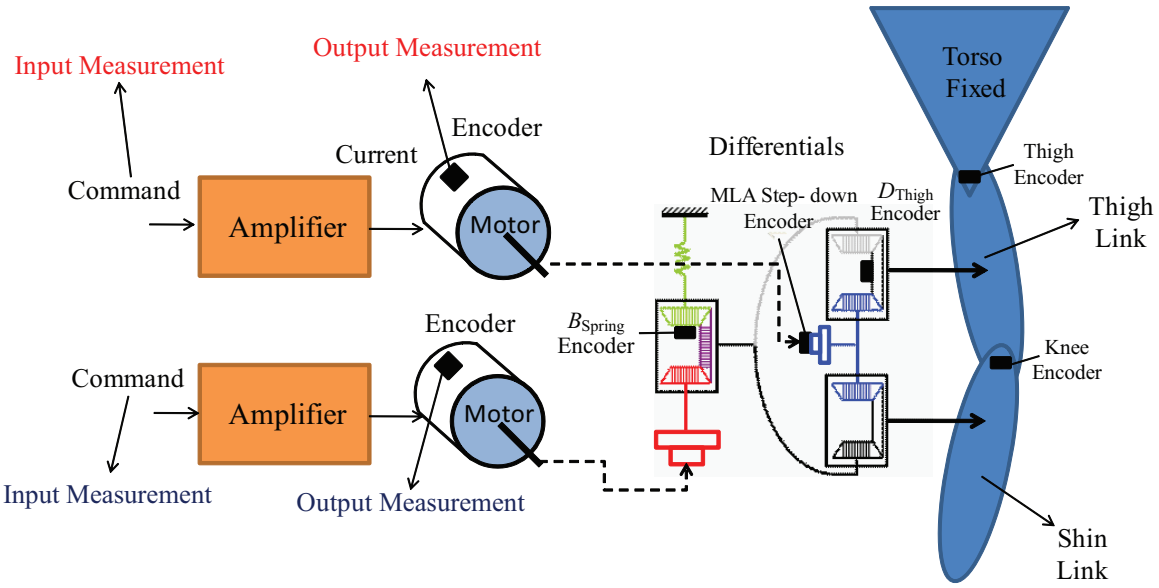


Figure 4.2: Experimental setup for parameter identification. Motor commands are logged as the input, while the encoder signals for the motor angles, the pulleys C_{Thigh} and D_{Thigh}, and the knee-joint angle are saved as outputs.

to the weight of the robot, and cable stretching occurs. This behavior is observed by the violation of the relations (2.3) and (2.4). Cable stretch is taken into account in the last step of the identification process.

It is typical for power amplifiers to exhibit a small bias in the commanded current, which in turn causes bias in the motor torques. Before beginning system identification, these biases are estimated and compensated for each motor following the procedure described in Appendix B.

4.4 Transmission Identification

For system identification, the fact that the differentials in the transmission are realized by a series of cables and pulleys is an advantage because we can select how many pulleys are actuated by disconnecting cables. For each pulley combination, the lumped moment of inertia can be computed. Therefore, if the electrical dynamics of the motor and power amplifiers are neglected, the lumped pulley system can be modeled as the first-order system

$$J_{\text{lumped}}\dot{\omega} + \mu_{\text{lumped}}\omega = u, \tag{4.1}$$

where J_{lumped} is the lumped moment of inertia, μ_{lumped} is the lumped friction coefficient, ω is the angular velocity of the motor, and u is the commanded motor torque. By identifying J_{lumped} and μ_{lumped} for three different combinations of pulleys plus motor, it is possible to determine K_{T} and J_{rotor}, as well as confirm the lumped pulley inertia predicted by the CAD model. For each side of the robot, the three pulley combinations of Figure 4.3(a) are used for the leg-angle path, while the three pulley combinations of Figure 4.3(b) are used for the leg-shape path.

4.4.1 Motor Torque Correction Factor and Inertia Correction Factor

The leg-angle-identification experiments are performed successively on the leg-angle motor in combination with one, three, and five pulleys as shown in Figure 4.3(a). The leg-shape-identification experiments are performed successively on the leg-shape motor in combination with one, three, and five pulleys as shown in Figure 4.3(b). The lumped moments of inertia of each combination, including the contributions of the cables, can be obtained by using (C.6) and (C.7) of Appendix C. The lumped moments of inertia of each combination, including the contributions of the cables can be expressed as

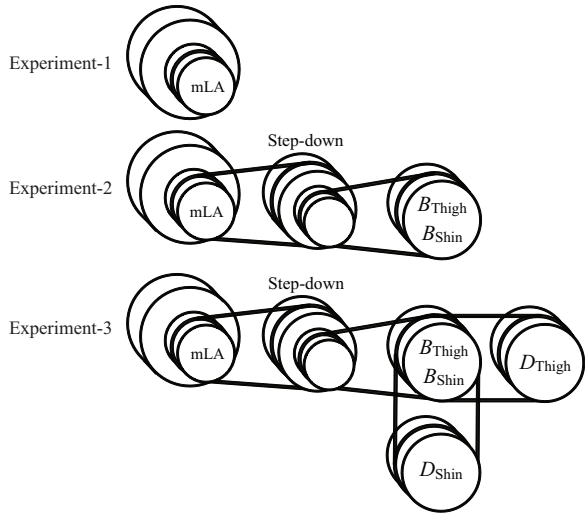
$$J_i = J_{\text{rotor}} + J_{\text{pulley},i} + J_{\text{cable},i}, \quad i = 1, 2, 3, \tag{4.2}$$

where i denotes experiment number, J_{rotor} is the inertia of the motor rotor, $J_{\text{pulley},i}$ is the lumped moment of inertia of the pulley combination for experiment i, which is obtained by combining the pulley inertias shown in Table 4.2 with the gear ratios between the pulleys taken into account, and $J_{\text{cable},i}$ is the lumped cable moment of inertia calculated from the mass of the cable per unit length and the length of the cable, with gear ratios taken into account.

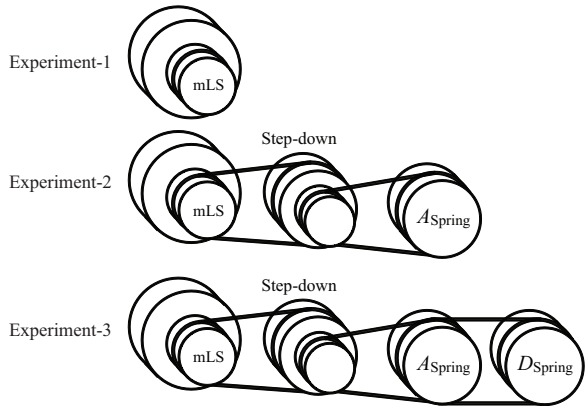
Letting $J_{\text{rotor,man}}$ denote the value of the rotor inertia supplied by the manufacturer, we define the scale factor α_{rotor} by

$$\alpha_{\text{rotor}} = \frac{J_{\text{rotor}}}{J_{\text{rotor,man}}}, \tag{4.3}$$

which we seek to estimate. In a similar manner, we define the scale factor α_{torque} for the



(a)



(b)

Figure 4.3: Pulley choices for identifying the transmission parameters. (a) q_{LA} path and (b) q_{LS} path. The various pulley combinations are formed by selectively disconnecting cables in the transmission. If the inertias of the pulleys are known, then two pulley combinations are sufficient to identify the inertia of the motor rotor and the torque constant. By adding a third pulley combination, the pulley inertias provided by the CAD program can be validated.

Table 4.2: Moments of inertia of the pulleys in the transmission as obtained from the CAD model.

Pulley	Moment of inertia (kg-m^2)
J_{mLS}	9.0144e-4
J_{mLA}	4.4928e-4
J_{Ath}	1.6680e-3
J_{Bth}	2.2181e-3
J_{Dth}	1.0826e-3
J_{Ash}	1.6974e-3
J_{Bsh}	2.2181e-3
J_{Dsh}	2.0542e-3
J_{Asp}	2.3464e-3
J_{Bsp}	1.8686e-3
J_{Dsp}	1.9313e-3
J_{mLSsd}	2.7117e-3
J_{mLAsd}	1.0950e-3

motor by

$$\alpha_{\mathrm{torque}} = \frac{K_{\mathrm{T}}}{K_{\mathrm{T,man}}}, \tag{4.4}$$

where $K_{\mathrm{T,man}}$ is the value of the motor torque constant supplied by the manufacturer and K_{T} is the true motor torque constant. In each experiment, the commanded motor torque is calculated by multiplying the current commanded by the power amplifier by $K_{\mathrm{T,man}}$. As illustrated in Figure 4.4, it follows that the transfer function from the commanded motor torque to the measured motor angular velocity is a scalar multiple of (4.1). Hence, the moment of inertia from the experiments is related to the moment of inertia of (4.2) by

$$J_{\mathrm{exp},i} = \alpha_{\mathrm{torque}}(\alpha_{\mathrm{rotor}} J_{\mathrm{rotor,man}} + J_{\mathrm{pulley},i} + J_{\mathrm{cable},i}), \quad i = 1, 2, 3, \tag{4.5}$$

where $J_{\mathrm{exp},i}$ is the lumped moment of inertia estimated on the basis of the ith experiment.

Three moment of inertia estimates, denoted by $J_{\mathrm{exp},1}$, $J_{\mathrm{exp},2}$, and $J_{\mathrm{exp},3}$ respectively, are obtained from each of the leg-shape and leg-angle experiments. Arranging the equations

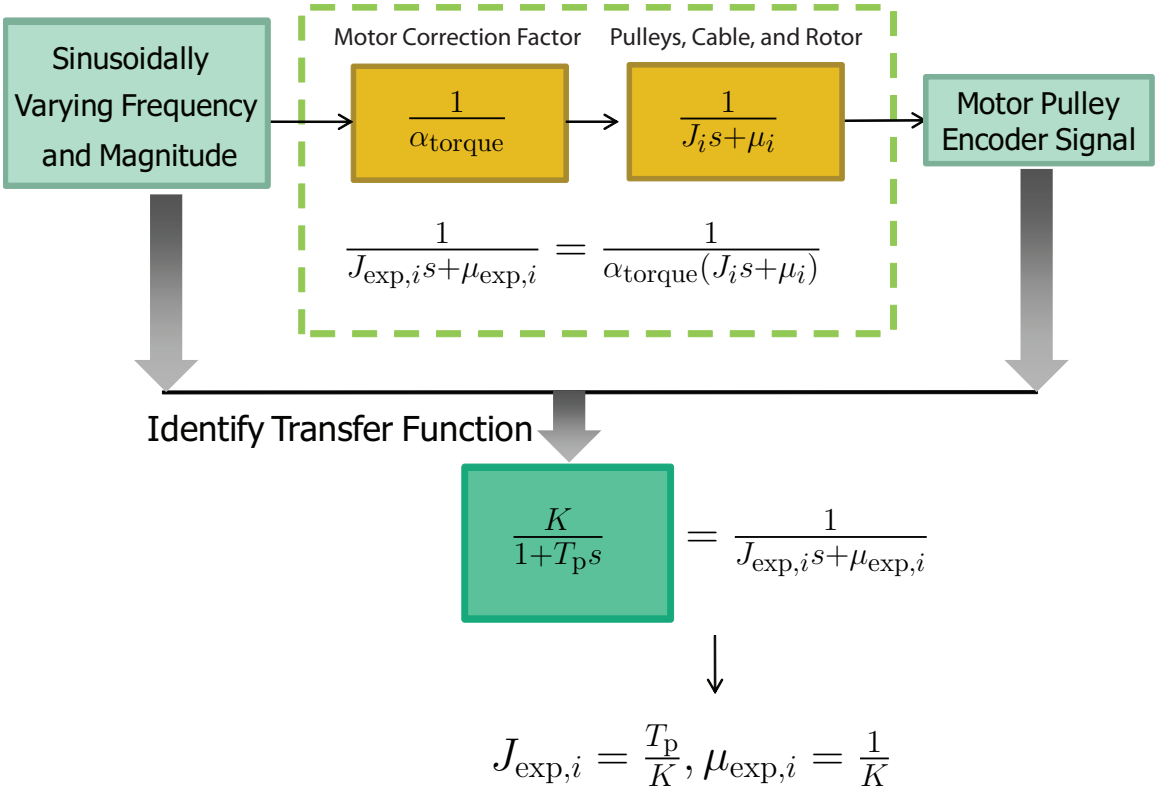


Figure 4.4: Transfer function from the the amplifier command input to the motor encoder signal output. The parameter α_{torque} is the correction factor for the motor torque constant supplied by the manufacturer. The measured transfer function is $1/\left[\alpha_{\text{torque}}(J_i s + \mu_i)\right]$. The Matlab System Identification Toolbox is used to estimate the first-order transfer function from the measured data.

for each set of experiments in matrix form gives

$$\Psi = \Gamma \begin{bmatrix} \alpha_{\text{torque}}\alpha_{\text{rotor}} \\ \alpha_{\text{torque}} \end{bmatrix}, \tag{4.6}$$

where

$$\Psi = \begin{bmatrix} J_{\text{exp},1} \\ J_{\text{exp},2} \\ J_{\text{exp},3} \end{bmatrix}, \quad \Gamma = \begin{bmatrix} J_{\text{rotor,man}} & J_{\text{pulley},1} + J_{\text{cable},1} \\ J_{\text{rotor,man}} & J_{\text{pulley},2} + J_{\text{cable},2} \\ J_{\text{rotor,man}} & J_{\text{pulley},3} + J_{\text{cable},3} \end{bmatrix}.$$

The estimates of α_{torque} and α_{rotor} are then obtained by least squares fit

$$\begin{bmatrix} \alpha_{\text{torque}}\alpha_{\text{rotor}} \\ \alpha_{\text{torque}} \end{bmatrix} \triangleq (\Gamma^T\Gamma)^{-1}\Gamma^T\Psi. \tag{4.7}$$

4.4.2 Experimental Results

Inputs for identification are designed as follows. Starting from 0.5 Hz, the input frequency is increased in 17 steps to 50 Hz. To allow the system response to reach steady state, each frequency is held constant for 10 periods until changing to the next higher frequency. At each frequency increment, the magnitude is also incremented to prevent the measured motor angular velocity from becoming too small. Figure 4.5 illustrates the input signal and corresponding system response. The Matlab System Identification Toolbox is used to identify the transfer function (4.1). Table 4.3 shows the results obtained from the experiments.

On the basis of the values in Table 4.3, α_{torque} and α_{rotor} are calculated by (4.7). The estimated values are listed in Table 4.4, along with the motor biases. The scale factors α_{rotor} indicate that the actual rotor inertias of the leg-shape motors are within 7% of the manufacturer's reported values, while the rotor inertias of the leg-angle motors are 25% less than the manufacturer's reported values. The scale factors α_{torque} of the leg-angle

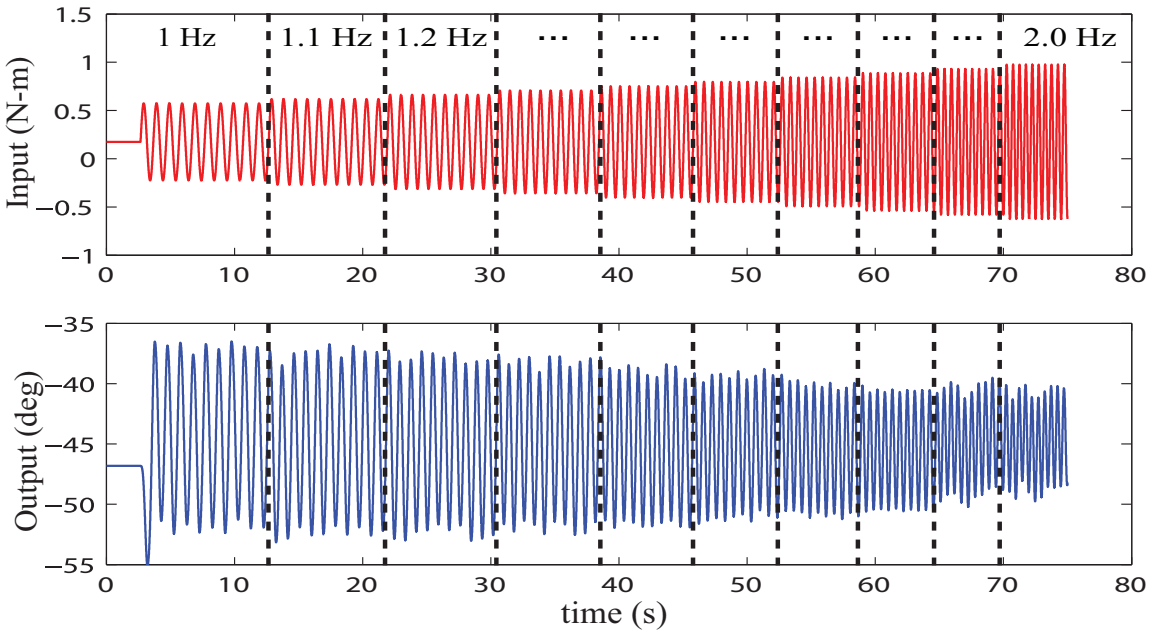


Figure 4.5: Illustrative input and output for system identification. The input is a modified chirp signal, that is, a sinusoid with time-varying frequency and magnitude. Each frequency is held constant until the system reaches steady state.

Table 4.3: Identified experimental moments of inertia and friction coefficients for the transmission mechanism.

		i=1	i=2	i=3
mLS_L	$J_{\text{exp},i}$ (kg-m^2)	8.819e-4	1.099e-3	1.112e-4
	$\mu_{\text{exp},i}$ (N-m-s/rad)	5.655e-3	6.518e-3	7.142e-3
mLA_L	$J_{\text{exp},i}$ (kg-m^2)	5.514e-4	7.223e-4	7.436e-4
	$\mu_{\text{exp},i}$ (N-m-s/rad)	2.332e-3	4.365e-3	3.858e-3
mLS_R	$J_{\text{exp},i}$ (kg-m^2)	1.104e-3	1.360e-3	1.431e-3
	$\mu_{\text{exp},i}$ (N-m-s/rad)	6.545e-3	9.811e-3	9.879e-3
mLA_R	$J_{\text{exp},i}$ (kg-m^2)	5.217e-4	6.900e-4	7.328e-4
	$\mu_{\text{exp},i}$ (N-m-s/rad)	1.718e-3	4.048e-3	4.703e-3

Table 4.4: Motor parameters. The rotor inertias $J_{\text{rotor,man}}$ and torque constants $K_{\text{T,man}}$ are provided by the manufacturer. The correction factors α_{rotor} and α_{torque} as well as the motor biases b are estimated from experimental data.

	mLS_L	mLA_L	mLS_R	mLA_R
$J_{\text{rotor,man}}$ (kg-m^2)	8.755e-4	4.880e-4	8.755e-4	4.880e-4
$K_{\text{T,man}}$ (N-m/A)	1.516	0.577	1.516	0.577
α_{rotor}	0.934	0.741	0.930	0.763
α_{torque}	0.995	1.332	1.287	1.269
b (N-m)	-0.1076	-0.04652	0.02995	-0.001672

motors on the left and right sides of the robot differ by less than 5%. For the leg-shape motors, the large difference in the scale factors α_{torque} is expected. In particular, motors of different characteristics for the left and right sides of the robot was necessary when one of the motors failed early in the construction process and was replaced with a motor from a previous prototype. We also note that motor biases, which range in magnitude from 0.002 to 0.107 N-m, are small in comparison to the torques that are expected in walking experiments, which can exceed 2 N-m for leg angle and 8 N-m for leg shape [100].

For the remainder of this chapter, the motor torque constant is computed by

$$K_{\text{T}} = \alpha_{\text{torque}} K_{\text{T,man}}.$$

4.5 Leg and Torso Parameters

4.5.1 Thigh and Shin Links

The thigh and shin links of the legs are actuated by the torque transmitted through the transmission, as shown in Figures 2.3(b) and 4.2. The total mass, center of mass, and inertia of each link are assumed known from the CAD model; their values are given in Table 4.5. The values of the motor torque constants and rotor inertias estimated for the transmission are assumed. Friction coefficients may differ from the values estimated for the transmission, however, because the hip and knee joints are now actuated. In this section, the torso continues to be fixed relative to the world frame and the position of the pulley B_{Spring} is fixed as well, removing compliance from the picture.

With the torso fixed, the left and right sides of the robot are, in principle, decoupled; in practice, some coupling of vibration from one side to the other occurs because the test stand is not perfectly rigid, but this coupling is ignored.

4.5.1.1 Mathematical Model

Because we assume rigid connections between q_{LA} and q_{mLA}, and between q_{LS} and q_{mLS} in the leg identification, the appropriate set of generalized coordinates for the dynamics of the combined leg and transmission systems is $q_g = [q_{\text{mLS}_\text{L}}, q_{\text{mLA}_\text{L}}, q_{\text{mLS}_\text{R}}, q_{\text{mLA}_\text{R}}]$.

Table 4.5: Mass, center of mass, and moment of inertia of the links obtained from the CAD model. The center of mass coordinates r_{x} and r_{y} for the robot are defined in Figure 2.3(a), while the moment of inertia is defined about the joint that passes through the origin of the corresponding link in Figure 2.3(a). For the boom, the coordinate r_{y} is measured from the central tower to the robot, while the coordinate r_{x} is zero.

Link	Mass (kg)	Center of Mass $[r_{\mathrm{x}}, r_{\mathrm{y}}]$ (m)	Moment of inertia (kg-m^2)	Length (m)
Spring (Csp)	1.8987	[0.0009406, 0.1181]	0.04377	-
Torso (T)	40.8953	[0.01229, 0.18337]	2.3727	-
Cshin (Csh)	1.6987	[0.0004345, 0.08684]	0.03223	-
Thigh (Th)	3.2818	[0.0003110, 0.1978]	0.1986	0.5
Shin (Sh)	1.5007	[0.0009671, 0.1570]	0.08813	0.5
Boom	7.2575	[0.0, 1.48494153]	20.4951	2.25

In the following, $\mathcal{Q} \subset \mathcal{I}$ in (2.5) represents coordinates of $\mathcal{I}$ that are actuated in a given experiment and will be called the actuated index set. Similarly, let $\mathcal{P} \subset \mathcal{I}$ be the complement of $\mathcal{Q}$; its elements correspond to the mechanically locked coordinates of $\mathcal{P}$, referred to as the locked index set. For example, suppose that only the $\mathrm{mLS_L}$ and $\mathrm{mLA_L}$ motor angles are actuated and the other coordinates are mechanically locked, then $\mathcal{Q} = \{\mathrm{mLS_L}, \mathrm{mLA_L}\}$ and $\mathcal{P} = \{\ \mathrm{mLS_R},\ \mathrm{mLA_R}\}$. We also define the set of coordinates $q_Q = \left[q_{\mathrm{q_1}}, \ldots, q_{\mathrm{q_{n_Q}}}\right]$, where $\mathrm{q_1}, \ldots, \mathrm{q_{n_Q}} \in \mathcal{Q}$, and $\mathrm{n_Q}$ is the number of elements in the set $\mathcal{Q}$, and $q_{\mathcal{P}} = \left[q_{\mathrm{p_1}}, \ldots, q_{\mathrm{p_{n_P}}}\right]$, where $\mathrm{p_1}, \ldots, \mathrm{p_{n_P}} \in \mathcal{P}$, and $\mathrm{n_P}$ is the number of elements in the set $\mathcal{P}$.

The parameters to be validated from the CAD model are grouped in a vector $\theta = [\mathbf{m} \quad \mathbf{I} \quad \mathbf{r_x} \quad \mathbf{r_y}]$, see Figure 2.3a, where $\mathbf{m}$, $\mathbf{I}$, $\mathbf{r_x}$ and $\mathbf{r_y}$ are mass, inertia, center of mass position in x, and in y, respectively (the values from the CAD model are presented in Table 4.5), and let $\alpha = [\alpha_{\mathrm{mLS_L}},\ \alpha_{\mathrm{mLA_L}},\ \alpha_{\mathrm{mLS_R}},\ \alpha_{\mathrm{mLA_R}}]$ from Table 4.4. The total kinetic energy for the actuated index set $\mathcal{Q}$ is

$$\begin{aligned} \mathcal{K}_Q\left(q_g, \dot{q_g}\, \theta, \alpha\right) =& \mathcal{K}^{Thigh}\left(q_g, \dot{q_g}, \theta, \alpha\right)|_{q_P = q_P^*} \\ &+ \mathcal{K}^{Shin}\left(q_g, \dot{q_g}, \theta, \alpha\right)|_{q_P = q_P^*} \\ &+ \mathcal{K}^{trans}\left(q_g, \dot{q_g}, \theta, \alpha\right)|_{q_P = q_P^*} \end{aligned} \tag{4.8}$$

where, $\mathcal{K}^{Thigh}$, $\mathcal{K}^{Shin}$, and $\mathcal{K}^{trans}$ are the kinetic energies of the thigh, the shin and the transmission, respectively, and q_P^* are the locked joint position angles for q_P. Symbolic

expressions for the transmission model are available online at [40]. The total potential energy for the actuated index set $\mathcal{Q}$ is

$$\begin{aligned}\mathcal{V}_Q\left(q_g, \dot{q_g}\,\theta, \alpha\right) =& \mathcal{V}^{Thigh}\left(q_g, \dot{q_g}, \theta, \alpha\right)|_{q_P=q_P^*} \\ &+\mathcal{V}^{Shin}\left(q_g, \dot{q_g}, \theta, \alpha\right)|_{q_P=q_P^*} \\ &+\mathcal{V}^{Trans}\left(q_g, \dot{q_g}, \theta, \alpha\right)|_{q_P=q_P^*}.\end{aligned} \tag{4.9}$$

The Lagrangian is then

$$\mathcal{L}_Q = \mathcal{K}_Q - \mathcal{V}_Q. \tag{4.10}$$

With the total kinetic energy and potential energy obtained from (4.8) and (4.9), the dynamics can be determined through Lagrange's equations:

$$\frac{d}{dt}\frac{\partial \mathcal{L}_Q}{\partial \dot{q}_Q} - \frac{\partial \mathcal{L}_Q}{\partial q_Q} = \Gamma_Q, \tag{4.11}$$

where Γ_Q is the vector of generalized forces acting on the robot, and can be written as:

$$\Gamma_Q = I_{\mathrm{n_q}\times\mathrm{n_q}} K_{T_Q} u_Q - F_\mu \dot{q}_Q, \tag{4.12}$$

where $I_{\mathrm{n_q}\times\mathrm{n_q}}$ is the identity matrix of size $\mathrm{n_q}$, $K_{T_Q} = diag[1/K_{T,\mathrm{q_1}} \cdots 1/K_{T,\mathrm{q_{n_p}}}]$, $u_Q = [\tau_{\mathrm{q_1}} + b_{\mathrm{q_1}} \cdots \tau_{\mathrm{q_{n_p}}} + b_{\mathrm{q_{n_p}}}]$, and $F_\mu = diag[\mu_{\mathrm{q_1}} \cdots \mu_{\mathrm{q_{n_p}}}]$, and where $K_{T,\mathrm{q_1 \cdots n_p}}$ are from Table 4.4. The $b_{\mathrm{q_1}}, \ldots, b_{\mathrm{q_{n_p}}}$ are the motor biases[1]. The motor biases can be obtained from Table 4.4 for the SISO experiment; for the MIMO experiment, however, they are obtained as part of the optimization process explained in next. The friction coefficients are $\mu_{\mathrm{q_1}}, \ldots, \mu_{\mathrm{q_{n_p}}}$, and the procedure to obtain them will also be explained in next.

The dynamic model for each side can be written in the form

$$D(\theta, \alpha_{\mathrm{rotor}}, q)\ddot{q} + C(\theta, \alpha_{\mathrm{rotor}}, q, \dot{q})\dot{q} + G(\theta, \alpha_{\mathrm{rotor}}, q) = \Gamma_Q, \tag{4.13}$$

where $D(\theta, \alpha_{\mathrm{rotor}}, q)$, $C(\theta, \alpha_{\mathrm{rotor}}, q, \dot{q})$, and $G(\theta, \alpha_{\mathrm{rotor}}, q)$ are the inertia matrix, Coriolis

[1]Because the legs are relatively light, small torque biases lead to significant errors in the modeled effects of gravity. When the robot is in actual operation and supporting the heavy torso, the effects of these small torque biases will be negligible.

matrix, and gravity vector, respectively. Moreover, θ is the vector of mechanical parameters from the CAD model, the rotor inertia correction factors α_{rotor} are from Table 4.4, and the vector Γ_Q of generalized forces acting on the robot, consisting of motor torque and viscous friction, is given by

$$\Gamma_Q = u_{\text{mLA}} + u_{\text{mLS}} - \mu_{\text{mLA}}\dot{q}_{\text{mLA}} - \mu_{\text{mLS}}\dot{q}_{\text{mLS}}. \tag{4.14}$$

The friction coefficients μ_{mLA} and μ_{mLS} are to be estimated.

Two types of experiment are performed, SISO and MIMO. Each experiment is performed on one leg at a time. In the SISO experiments, one degree of freedom is actuated and logged, either q_{mLS} or q_{mLA}, while the other degree of freedom is mechanically locked. In the MIMO experiment, both q_{mLS} and q_{mLA} are actuated and recorded. The objective of the SISO experiments is to estimate the friction parameters in (4.14). The objective of the MIMO experiment is to validate the model (4.13), with the parameters obtained from the SISO experiment.

The commanded motor torque is a modified chirp signal plus a constant offset, similar to the transmission identification experiments. The magnitude and offset of the input signal must be selected to keep the links within the robot's work space.

With all the parameters in the model (4.13) specified, the response of the system excited by the input used in the experiments can be simulated, as shown in Figure 4.6. The friction parameters μ are estimated by minimizing the cost function

$$J(\mu) = \sqrt{\sum(y_{\text{exp}} - y_{\text{sim}}(\mu))}, \tag{4.15}$$

where y_{exp} is the vector of experimentally measured data, y_{sim} is the vector of simulated data, and μ is the vector of viscous-friction coefficients. As shown in Table 4.6, the values of μ estimated in this manner are larger than the values from the transmission experiments, but not greatly different from those values. In the MIMO simulations, we observe that variations in the assumed actuator bias of 0.1 N-m, which can be ignored when the legs are supporting the robot, can cause large deviations in the system response when the legs

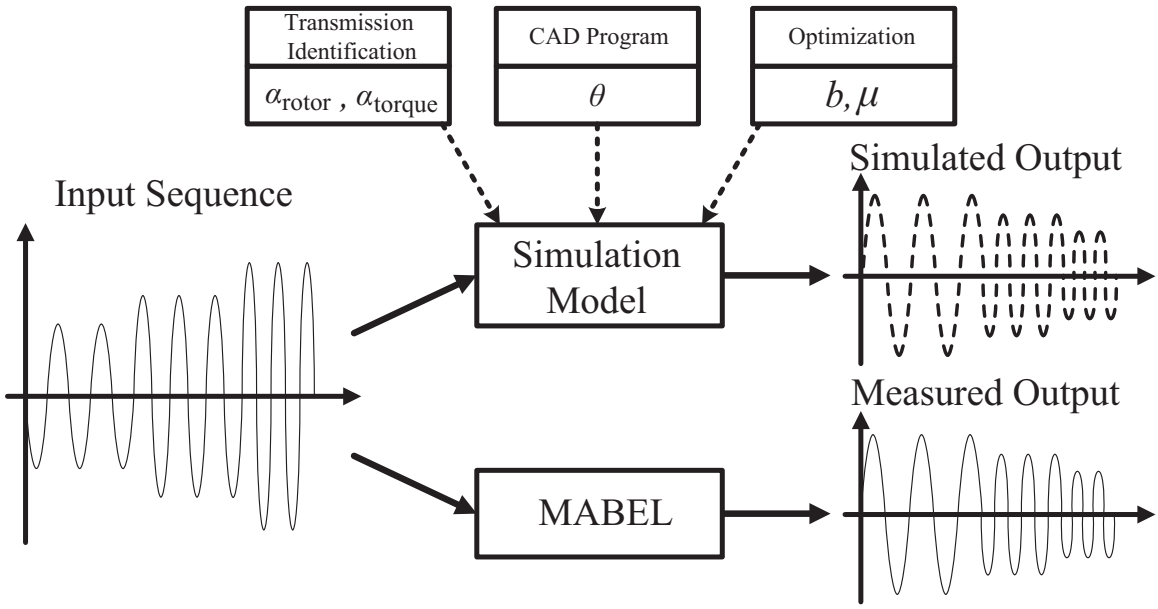


Figure 4.6: Simulation and validation procedures for leg identification. Identical input sequences are applied to the simulation model and MABEL. The simulation model uses the motor parameters estimated from the transmission identification along with the mass-inertia parameters calculated from the CAD program, while torque bias and friction coefficients are free variables used to fit the simulation data to the experimental data. In the validation step, a new input sequence is used and the motor bias is re-estimated, while all remaining parameters are held constant.

Table 4.6: Estimates of friction coefficients μ and motor biases b obtained by minimizing the costs in (4.15) and (4.16), respectively.

	i = mLS_L	i = mLA_L	i = mLS_R	i = mLA_R
μ_i (N-m-s/rad)	9.844e-3	4.316e-3	9.027e-3	4.615e-3
b_i (N-m)	-8.417e-3	2.597e-2	-1.446e-2	-2.461e-3

are not supporting the robot. Therefore, for the MIMO simulations, in place of the bias values estimated from the transmission identification, we use the values that minimize the cost function

$$J(b) = \sqrt{\sum (y_{\text{exp}} - y_{\text{sim}}(b))}, \tag{4.16}$$

where y_{exp} is the vector of experimentally measured data, y_{sim} is the vector of simulated data, and b is the bias vector.

The comparisons between simulated and experimental results are presented in Figures 4.7 and 4.8. All figures show q_{LS} and q_{LA} computed from q_{mLS} and q_{mLA}, because the body coordinates q_{LS} and q_{LA} are easier to interpret than the motor positions. It is emphasized that the parameters are either from the transmission identification experiments or the CAD model, with two exceptions, namely, friction is estimated in the SISO experiments from (4.15) and used in the MIMO experiments; in the MIMO experiments, motor biases are tuned by (4.16).

The comparison of the MIMO experimental and simulation results is made further in Figures 4.9 and 4.10, where the phase portraits of q_{LA} versus q_{LS} with respect to each frequency component are plotted, for each side of the robot. We can observe that simulation results closely match the experimental results.

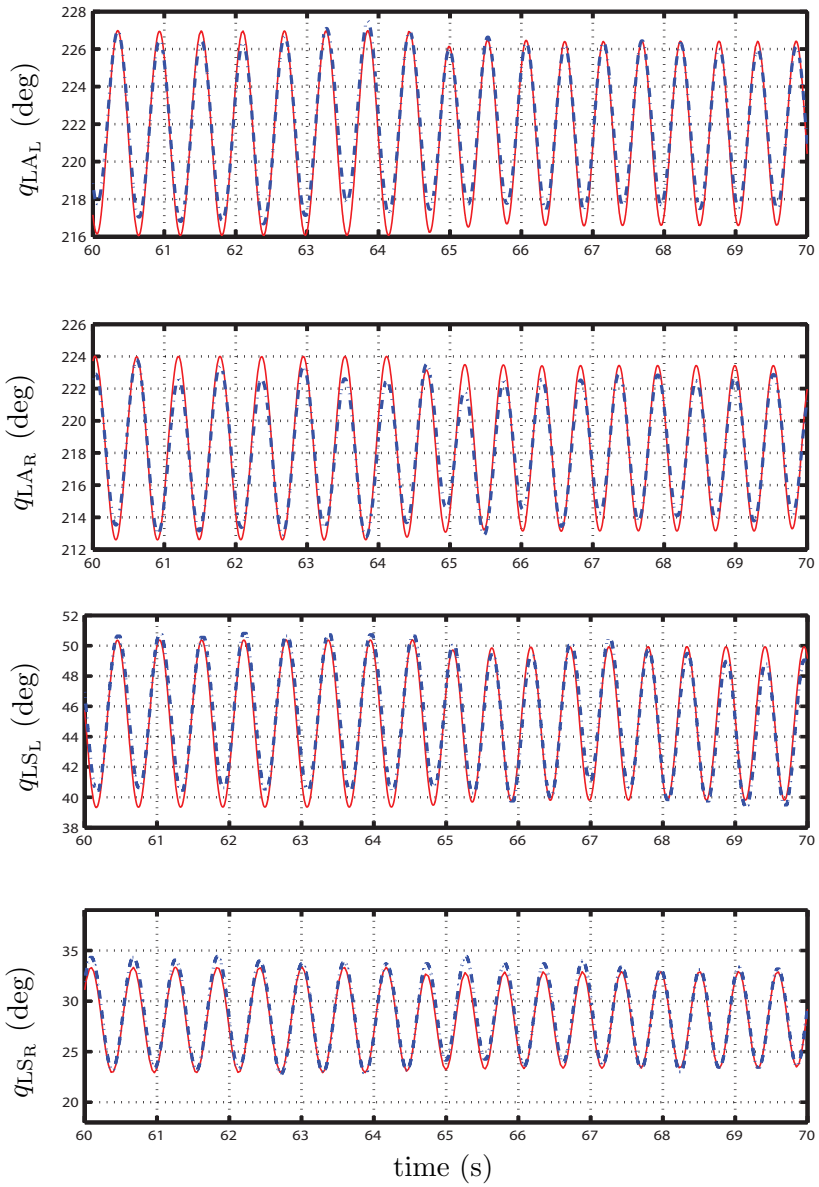


Figure 4.7: SISO simulation and experimental data. The joint positions for the simulation are indicated by solid red lines, while the joint positions for the experiments are indicated by dotted blue lines. These data are obtained from the procedure in Figure 4.6. The RMS errors are 0.77, 1.1, 0.76, and 0.69 deg, respectively.

In the SISO experiments, the RMS error varies from 0.69 to 1.1 deg, while for the MIMO experiments, the RMS error varies from 0.72 to 1.42 deg. These errors may arise from several sources. For example, the linear viscous-friction terms in (4.14), namely $\mu_{\mathrm{mLA}}\ \dot{q}_{\mathrm{mLA}}$ and $\mu_{\mathrm{mLS}}\ \dot{q}_{\mathrm{mLS}}$, do not take into account stick-slip behavior in the low-velocity region. Furthermore, electrical wiring is not included in calculating inertial parameters. In addition, motor bias changes slightly for each experimental trial. Finally, cable stretch is assumed to be negligible.

4.5.2 Torso

The torso represents approximately 41 kg of the total 65-kg mass of the robot. Consequently, the mass and inertia of the torso strongly affect the dynamics of the robot. In principle, the inertia and mass distribution of the torso can be validated by locking the legs in a fixed position, and using the leg-angle motors to oscillate the torso. Attempts at executing this experiment in the test stand failed, since movement of the torso is always translated to the legs.

Therefore, instead of dynamic identification of the torso, static balancing experiments are used to validate the CAD model estimates. In the first experiments, the robot is not constrained. Using local PD controllers, we command a posture where the right leg is extended more than the left leg. The robot is then balanced by hand on the right leg. The balance of the robot is maintained with minimal fingertip pressure from one of the experimenters. Once the robot is in a balanced posture, the joint position data are recorded. Many different postures are balanced and logged. From the logged data, we calculate the center of mass position of the robot including the horizonal boom, and we verify that the calculated center of mass is located over the supporting toe.

In a second set of experiments, the position of the hip joint is fixed, with the legs hanging below the robot and above the floor, and with the robot unpowered. The torso is balanced by hand in the upright position. We then calculate the center of mass position of the robot without the boom, and check that the center of mass is aligned over the hip joint.

We use 10 different postures for the first experiment and 7 different postures for the second experiment. Figure 4.11(a) displays the horizontal distance between the center

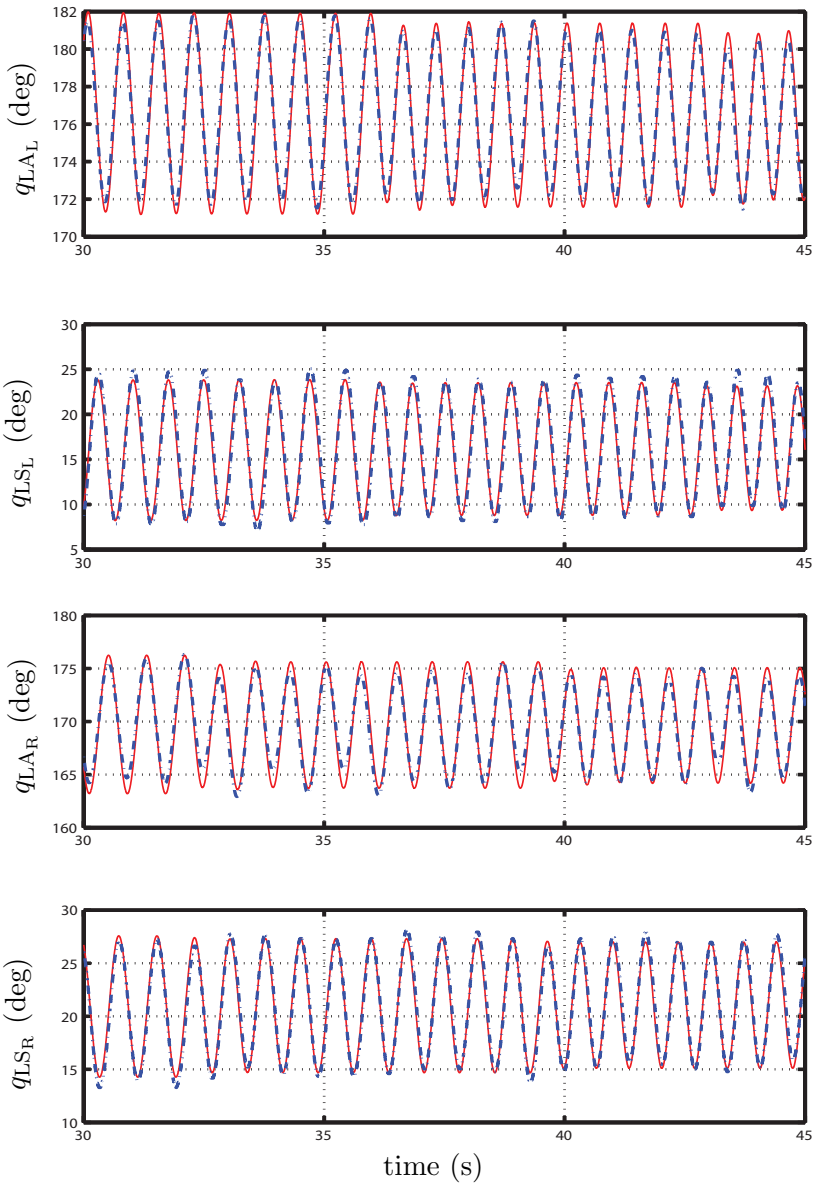
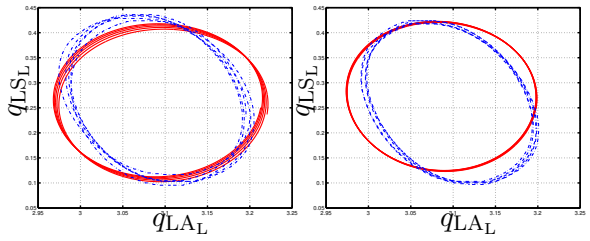
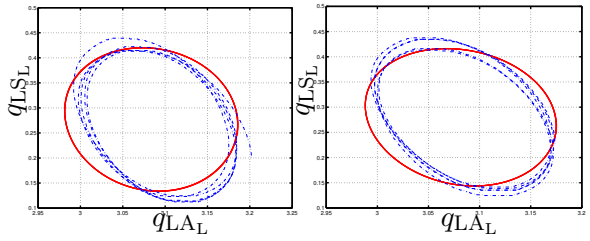


Figure 4.8: MIMO simulation and experimental data. The joint positions for the simulation are indicated by solid red lines, while the joint positions for the experiments are indicated by dotted blue lines. These data are obtained from the procedure in Figure 4.6. The RMS errors are 1.0, 1.42, 1.14, and 0.72 deg, respectively.



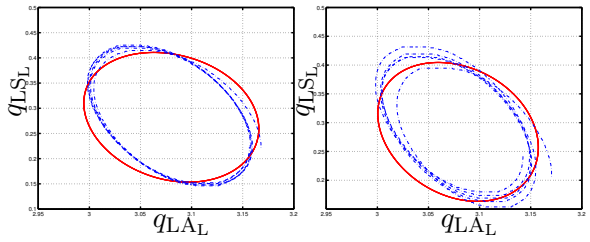
(a) 1.0000 Hz

(b) 1.0801 Hz



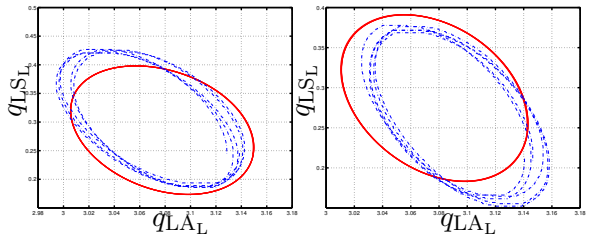
(c) 1.1665 Hz

(d) 1.2599 Hz



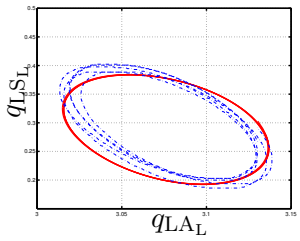
(e) 1.3608 Hz

(f) 1.4697 Hz



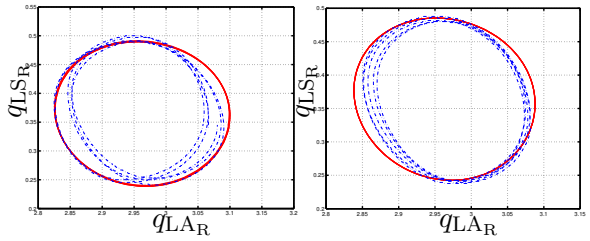
(g) 1.5874 Hz

(h) 1.7145 Hz



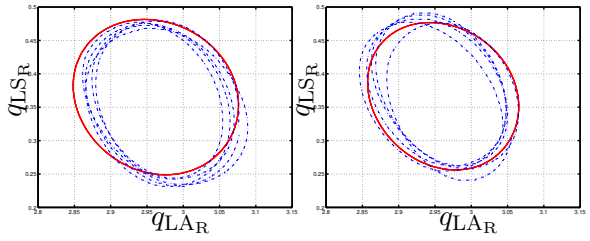
(i) 1.8517 Hz

Figure 4.9: MIMO phase plot of q_{LA_L} vs q_{LS_L} simulations (solid red line) and experiments (dotted blue line). All data are measured in radians.



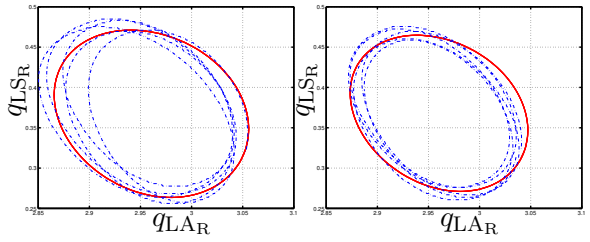
(a) 1.0000 Hz

(b) 1.0801 Hz



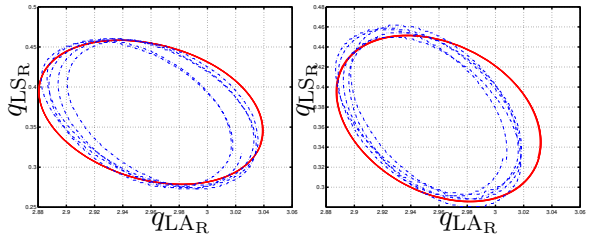
(c) 1.1665 Hz

(d) 1.2599 Hz



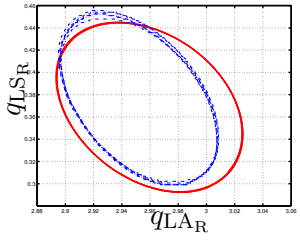
(e) 1.3608 Hz

(f) 1.4697 Hz



(g) 1.5874 Hz

(h) 1.7145 Hz



(i) 1.8517 Hz

Figure 4.10: MIMO phase plot of q_{LA_R} vs q_{LS_R} simulations (solid red line) and experiments (dotted blue line). All data are measured in radians.

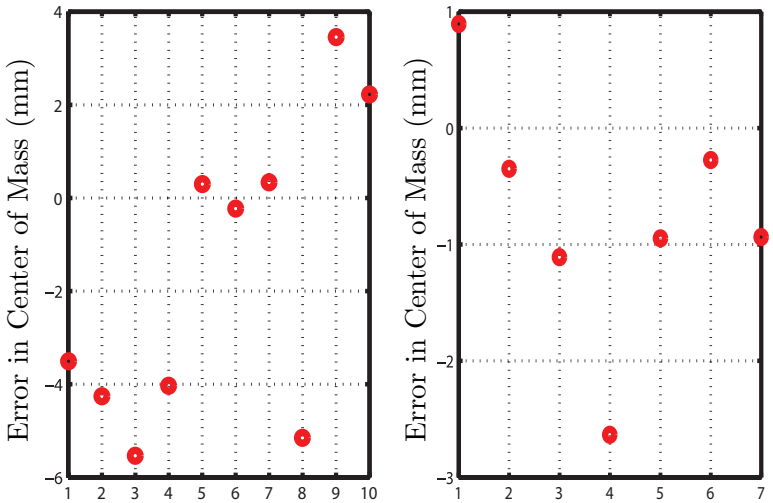


Figure 4.11: Errors in the estimated horizontal position of the center of mass. (a) In these experiments, the robot is manually balanced on one leg, in which case the center of mass of the robot plus the boom must be over the toe. The CAD model provides an estimate of the center of mass. The graph depicts the difference in the measured horizonal component of the center of mass and the CAD-model estimate for ten postures of the robot. The maximum difference is less than 6 mm. (b) In these experiments, the cartesian position of the hip is clamped to a fixed position, with the legs extended below the robot and off the floor. The torso is then balanced in an upright position, providing the horizontal position of the center of mass of the torso. This plot depicts the measured error with respect to the CAD-model estimate for seven trials of the experiment.

of mass and the supporting toe for the first experiment, and Figure 4.11(b) shows the horizontal distance between the center of mass and the hip for the second experiment. We observe that the maximum error is 6 mm, which is negligible in view of the size of the robot and considering that the experiments are performed with manual balancing. These experiments do not provide information on the vertical position of the center of mass.

4.6 Compliance

The stiffness K_{B} of the springs in (3.19) that are in series with the leg-shape actuators is estimated by means of static experiments using the calculated spring torques and measured spring deflections in this section. The remaining parameters Kd_{B}, K_{d1}, K_{vd1}, and K_{vd2} in (3.19) will be estimated in Section 4.7.

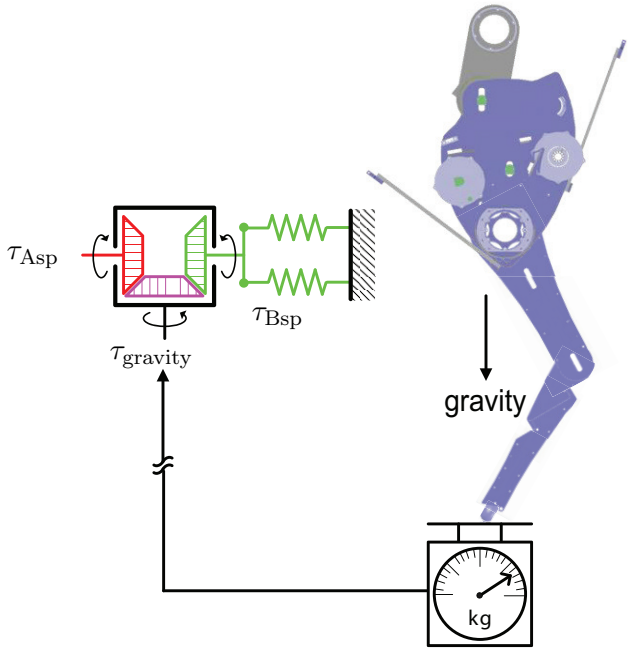


Figure 4.12: Experimental setup for measuring spring stiffness and pre-load. Pulley Asp is locked in place by commanding the leg-shape motor to a fixed orientation. Consequently, the torque due to gravity τ_{gravity} must be balanced by the torque from the spring τ_{Bsp}. The resulting spring deflection q_{Bsp} is measured and recorded.

The magnitude of the joint torques used in these experiments is more representative of the torques used in walking [100]. Under these greater loads, the cables in the differentials stretch. This compliance has been modeled in Section 3.2.2, and the spring and damping coefficients of the cables are identified in Section 4.7.

The series compliance in the drivetrain is now estimated by means of static, constant-torque experiments, performed by balancing the robot on one leg at a time. The setup is illustrated in Figure 4.12. In these experiments, the torso is no longer locked in place relative to the world frame; it is free. The actuators on one side of the robot are disabled, with the leg on that side folded and tied to the torso. On the other side, a PD-controller is used to maintain the leg angle at 180 deg, which is straight down. A second PD-controller is used to set the nominal leg shape. An experimenter balances the robot in place with the toe resting on a scale placed on the floor; the experimenter adjusts the overall angle of the robot so that it is exactly balanced on the toe, as when verifying the center of mass of the torso.

With the setup shown in Figure 4.12, the scale is measuring the combined weight of the

robot and the boom. By the design of the differential, when the leg-shape motor holds the pulley A_{Spring} in a fixed orientation, the torque at the pulley C_{Spring} is exactly balanced by the torque at the pulley B_{Spring}. The torque τ_{gravity} at C_{Spring}, shown in Figure 4.12, is the weight of the robot transmitted through the thigh and shin differentials; its magnitude is given by

$$|\tau_{\text{gravity}}| = \left| \frac{1}{2} W_{\text{robot}} \sin(q_{\text{LS}}) \right|, \tag{4.17}$$

where W_{robot} is the weight of the robot measured by the scale at the bottom of the foot. The absolute value is used because spring stiffness is positive. The torque τ_{Bsp} at the pulley B_{Spring}, shown in Figure 4.12, is due to the deflection of the spring and is given by

$$\tau_{\text{Bsp}} = K_{\text{B}} q_{\text{Bsp}}, \tag{4.18}$$

where K_{B} is the spring stiffness. The spring deflection q_{Bsp} is measured by an encoder installed in the pulley B_{Spring}. The torques τ_{gravity} and τ_{Bsp}, which are related by the differential mechanism, satisfy

$$|\tau_{\text{gravity}}| = 2.59 \, |\tau_{\text{Bsp}}| \, . \tag{4.19}$$

Combining (4.17), (4.18), and (4.19), the spring stiffness is obtained as

$$K_{\text{B}} = \frac{1}{5.18} \left| \frac{W_{\text{robot}} \sin(q_{\text{LS}})}{q_{\text{Bsp}}} \right| . \tag{4.20}$$

The design of the experiment is completed by varying q_{LS} over a range of values, here taken to be from 10 deg to 30 deg. We emphasize that K_{B} determined by means of (4.20) does not depend on the estimated leg-shape motor torque. Indeed, because the spring and motor are connected in series, the torques must balance at the spring differential.

Figure 4.13 shows the results of performing the above experiments on each leg. Prior to the experiments, it is not obvious if a linear model of the spring deflection would be adequate because the spring folds around the curved torso as shown in Figure 4.14. From the data, we observe that the spring behavior is nearly linear, and that the spring constants

for the left and right sides are consistent.

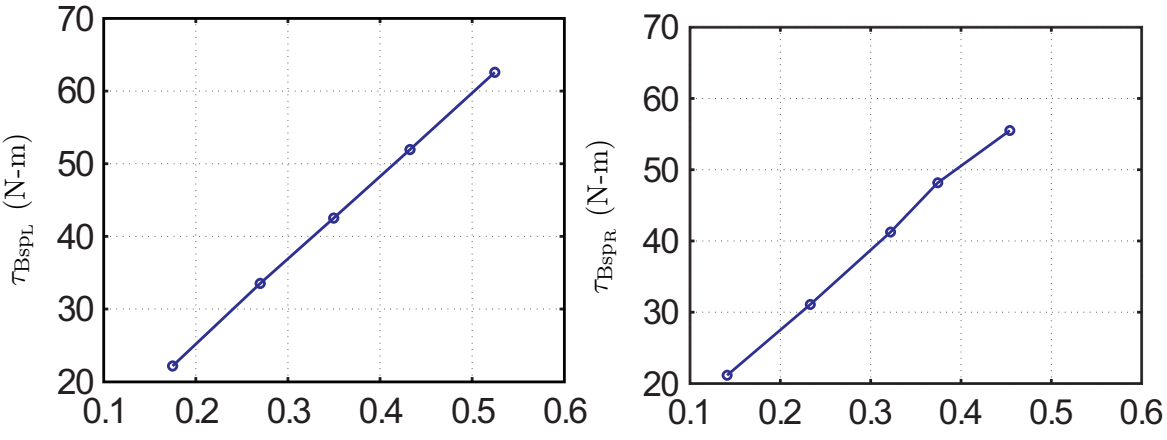
4.7 Two-Legged Hopping for Final Model Identification and Validation

This section uses a hopping gait to complete the identification of, and subsequently validate, the overall dynamic model of MABEL. While hopping *per se* is not an objective, the large motor torques employed to launch the robot in the air, in combination with the large ground contact forces at landing, make hopping a convenient means for exciting all of the dynamic modes that are expected to be present when the robot is running on flat ground or walking on uneven ground. Two-legged hopping is chosen over hopping on one leg to simplify the task of obtaining stable hopping.

4.7.1 Parameters for Cable Stretch, Hard Stop, and Ground Models

A heuristic controller for a hopping gait is given in Appendix D. Because the cable stretch coefficients are not yet identified, the hopping controller is hand tuned on an approximate simulation model that assumes the cables are rigid; the approximate model also uses the ground reaction parameters from [80]. The tuning process adjusts PD-controller gains and setpoints in the various phases of the gait, with the goal of obtaining sustained hopping. When the controller is implemented on MABEL, steady-state hopping is not achieved, even after extensive trial-and-error tuning in the laboratory. Five hops is typical before the robot falls.

Even though sustained hopping is not achieved, the data from the experiments can be used to identify the parameters in the hard-stop model, the cable stretch model, and the compliant-ground-contact model. The parameter fitting is accomplished with a combination of hand adjustment and nonlinear least squares. The resulting parameters are given in Table 4.7. Figures 4.15, 4.16, and 4.17 compare the result of the experiment (dotted blue line) and the simulation model (solid red line) using the parameters in the TABLE 4.7. Excellent agreement was obtained.



(a) Left Spring

(b) Right Spring

Figure 4.13: Measurements of torque τ_{Bsp} versus displacement q_{Bsp} obtained from the experimental setup of Figure 4.12. The data in (a) are fit by $\tau_{\mathrm{Bsp_L}} = 115.1 q_{\mathrm{Bsp_L}} + 2.214$. The fit shows that, on the left side of the robot, the spring stiffness is 115.1 N-m per rad and the pre-load offset is 2.214 N-m. On the right side of the robot, the data in (b) are fit by $\tau_{\mathrm{Bsp_R}} = 111.7 q_{\mathrm{Bsp_R}} + 5.377$, showing that the spring stiffness is 111.7 N-m per rad and the pre-load offset is 5.377 N-m. The estimated values of spring stiffness and pre-load are consistent on the left and right sides of the robot.

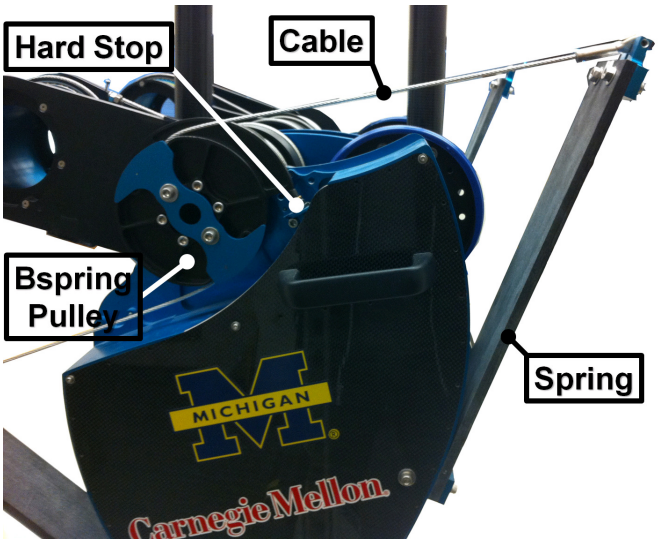


Figure 4.14: MABEL's unilateral springs. The cable, hard stops, and pulley B_{Spring} are also shown. The spring, fabricated from fiberglass plates, is activated when the pulley B_{Spring} rotates and pulls the cable, wrapping the fiberglass plates around the front and back of the torso. The cable cannot push the spring, and hence the spring is unilateral. A rest position is enforced by a hard stop, formed by a stiff spring-damper.

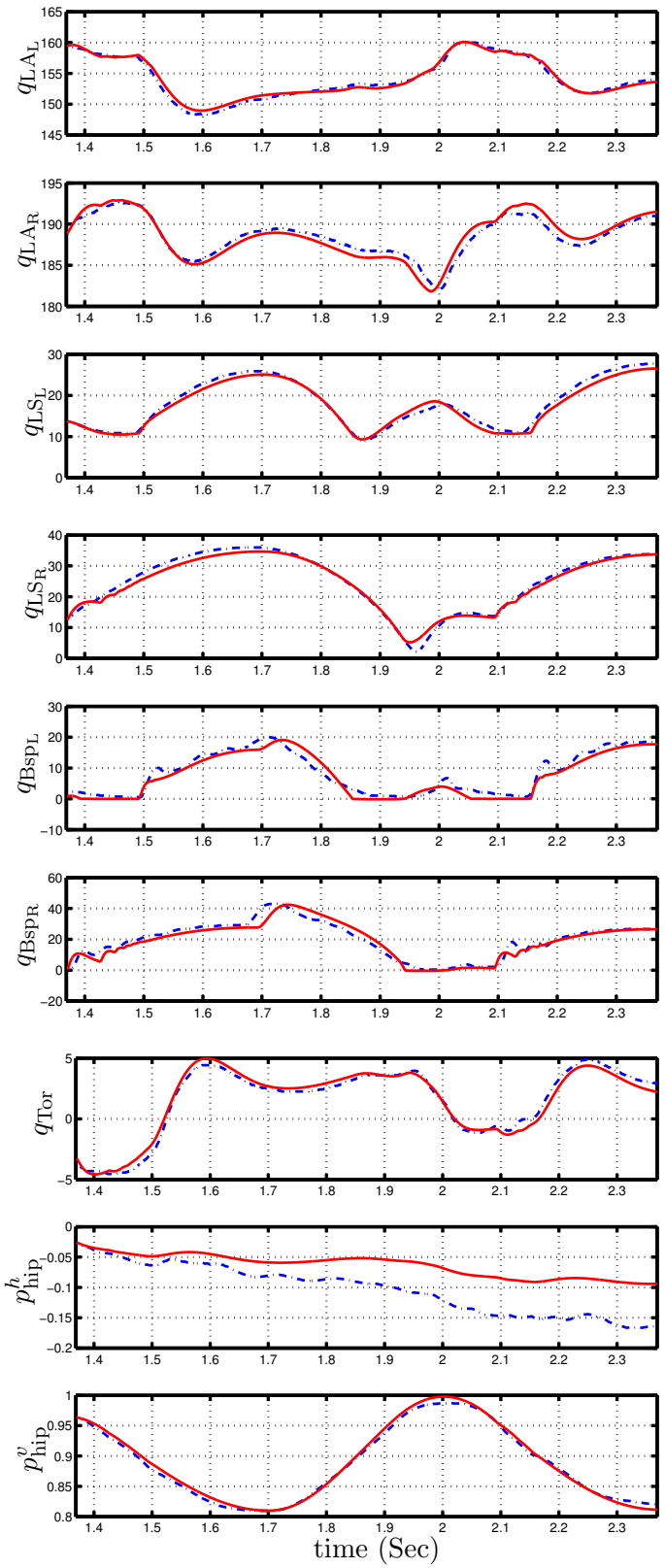


Figure 4.15: Identification data from the intimal hopping experiment. Joint position (in deg) and hip position (in m): simulation (solid red line) and experiment (dotted blue line).

Table 4.7: Parameters estimated from the dynamic hopping experiment.

Spring Model			
Kd_{B} (N-m-s/rad)	1.5	K_{vd1} (N-m-s/rad)	1000
K_{d1} (N-m/rad^3)	100	K_{vd2} (N-m/(rad/s)$^{0.5}$)	50

Cable Stretch Model				
	i = mLS$_{\mathrm{L}}$	i = mLA$_{\mathrm{L}}$	i = mLS$_{\mathrm{R}}$	i = mLA$_{\mathrm{R}}$
$K_{\mathrm{C,i}}$ (N-m/rad)	2.9565	3.5000	2.9565	3.8094
$Kd_{\mathrm{C,i}}$ (N-m-s/rad)	0.0402	0.0889	0.0804	0.3556

Ground Model			
λ_{v}^{a} (N/m$^{1.5}$/(m/s))	3.0e6	σ_{h0} (N/m)	260.0
λ_{v}^{b} (N/m$^{1.5}$/(m/s)$^{0.5}$)	4.5e6	σ_{h1} (N-s/m)	2.25
n	1.5	α_{h0} (N)	1.71
k (N/m$^{1.5}$)	4.38e7	α_{h2} (N-s/m)	0.54

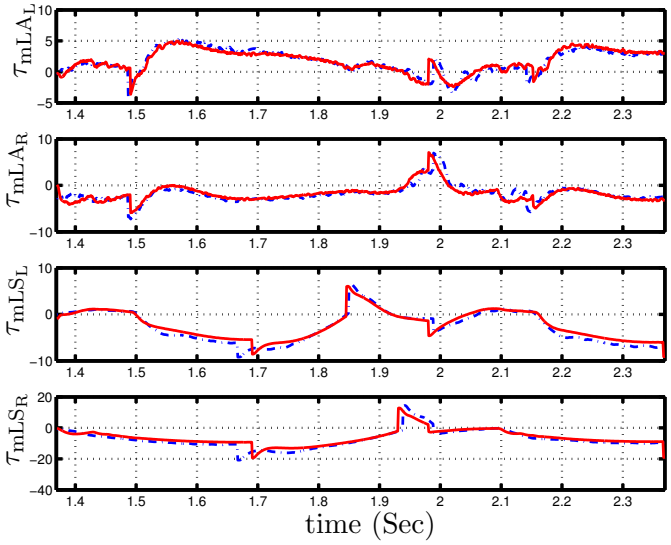


Figure 4.16: Identification data from the intimal hopping experiment. Joint torque (in Nm): simulation (solid red line) and experiment (dotted blue line).

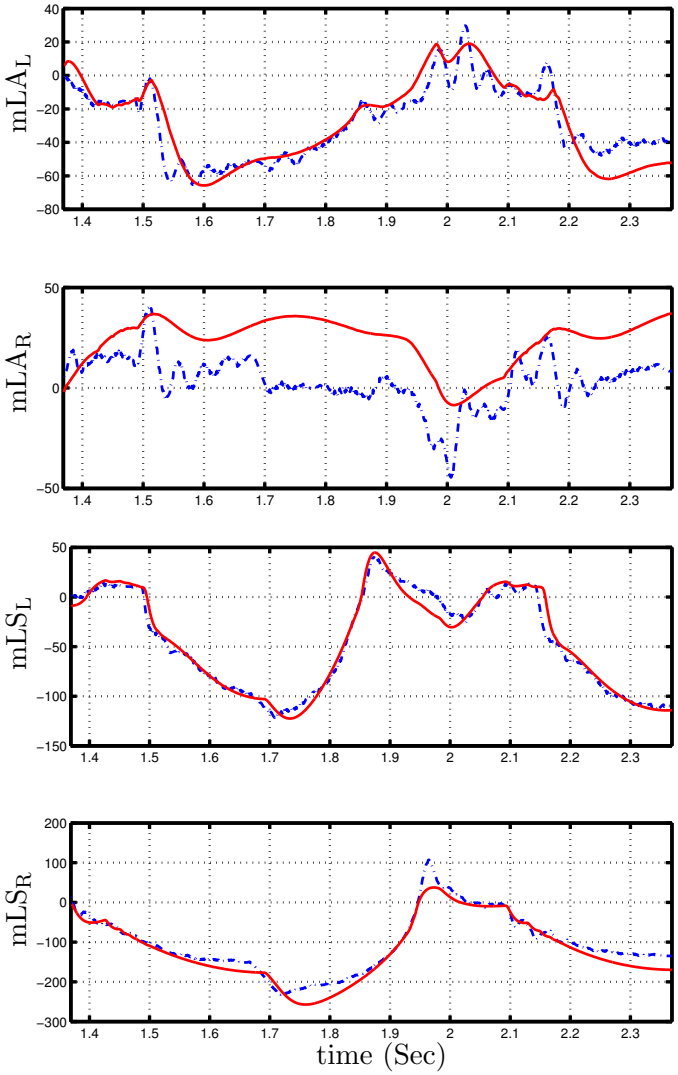


Figure 4.17: Identification data from the intimal hopping experiment. Cable stretch (in deg): simulation (solid red line) and experiment (dotted blue line).

4.7.2 Hopping Experiments for Validation

When the nominal hopping controller is simulated on the identified model, the closed-loop system is found to be unstable. Event-based updates to the torso angle are therefore added to achieve stability, as explained in "How Hopping is Achieved." The controller is then applied to MABEL, resulting in 92 hops before the test is deliberately terminated. The data are presented next.

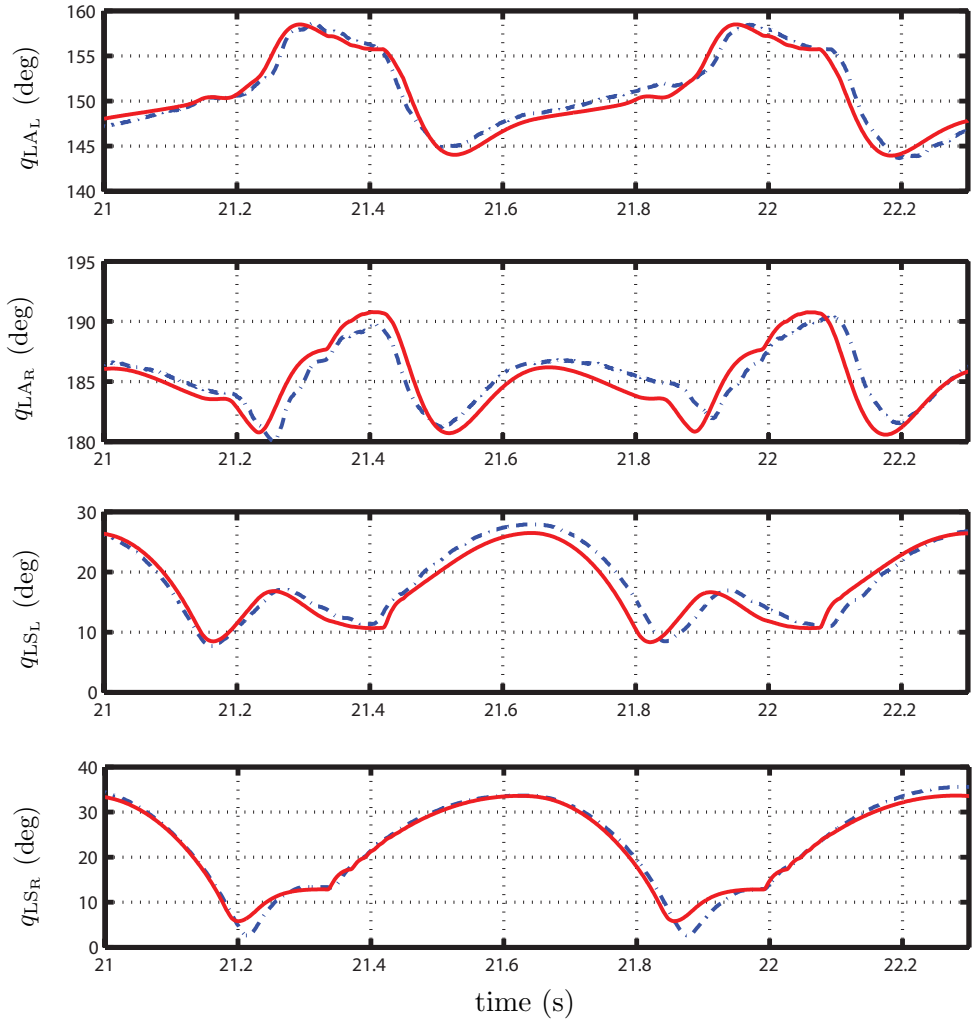


Figure 4.18: Validation data from the second hopping experiment. The joint positions for the simulation are indicated by solid red lines, while the joint positions for the experiment are indicated by dotted blue lines. Though the hopping controller is not based on trajectory tracking, the variables shown are indirectly regulated by the feedback controller, which may be responsible for the close correspondence between the modeled and measured data.

Figures 4.18 - 4.21 compare typical experimental results against the simulation results

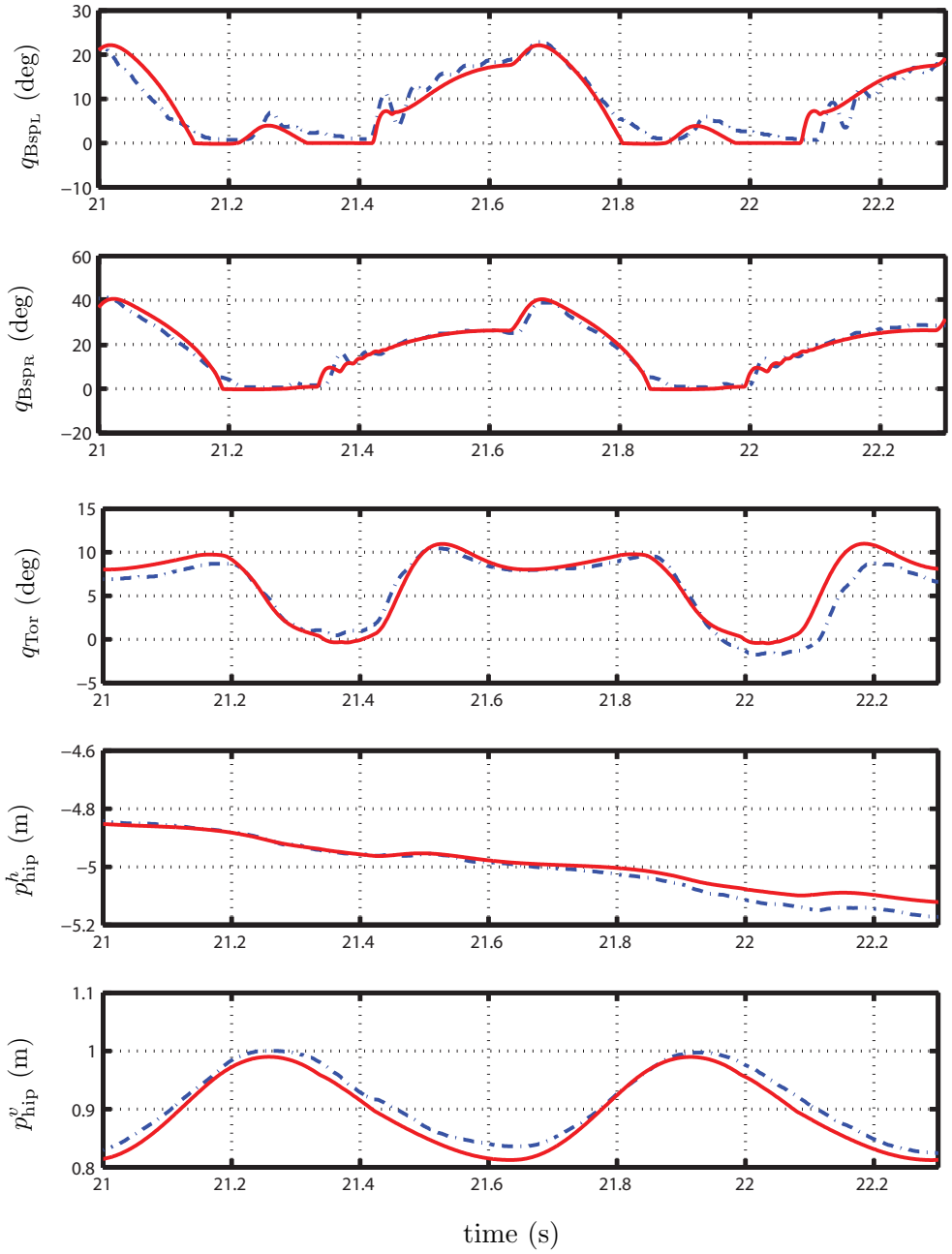


Figure 4.19: Validation data from the second hopping experiment. The joint and hip positions for the simulation are indicated by solid red lines, while the joint and hip positions for the experiment are indicated by dotted blue lines. For these variables, the close correspondence between the modeled and the measured data supports the model.

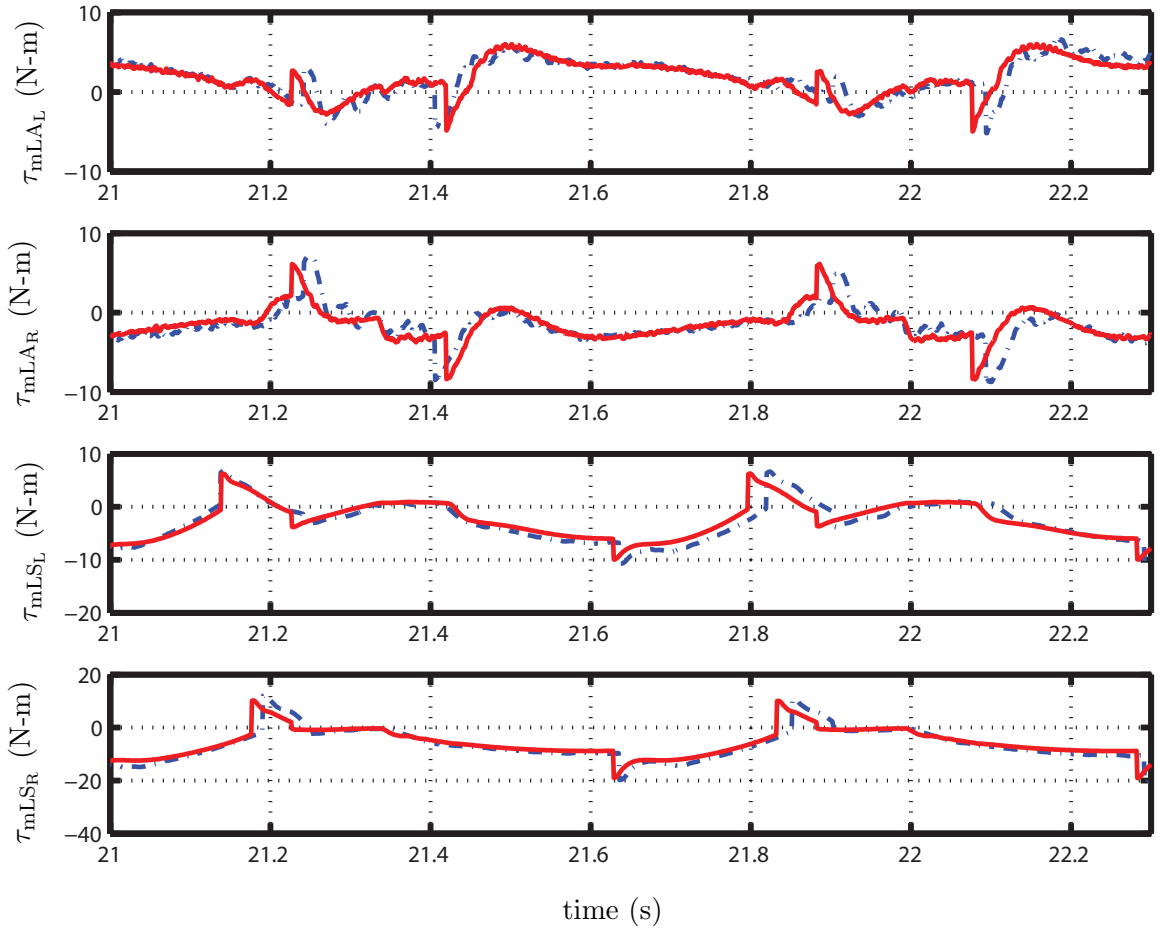


Figure 4.20: Validation data from the second hopping experiment. The motor torques for the simulation are indicated by solid red lines, while the motor torques for the experiment are indicated by dotted blue lines. The ability to predict torque is an accepted measure of fit for models of mechanical systems.

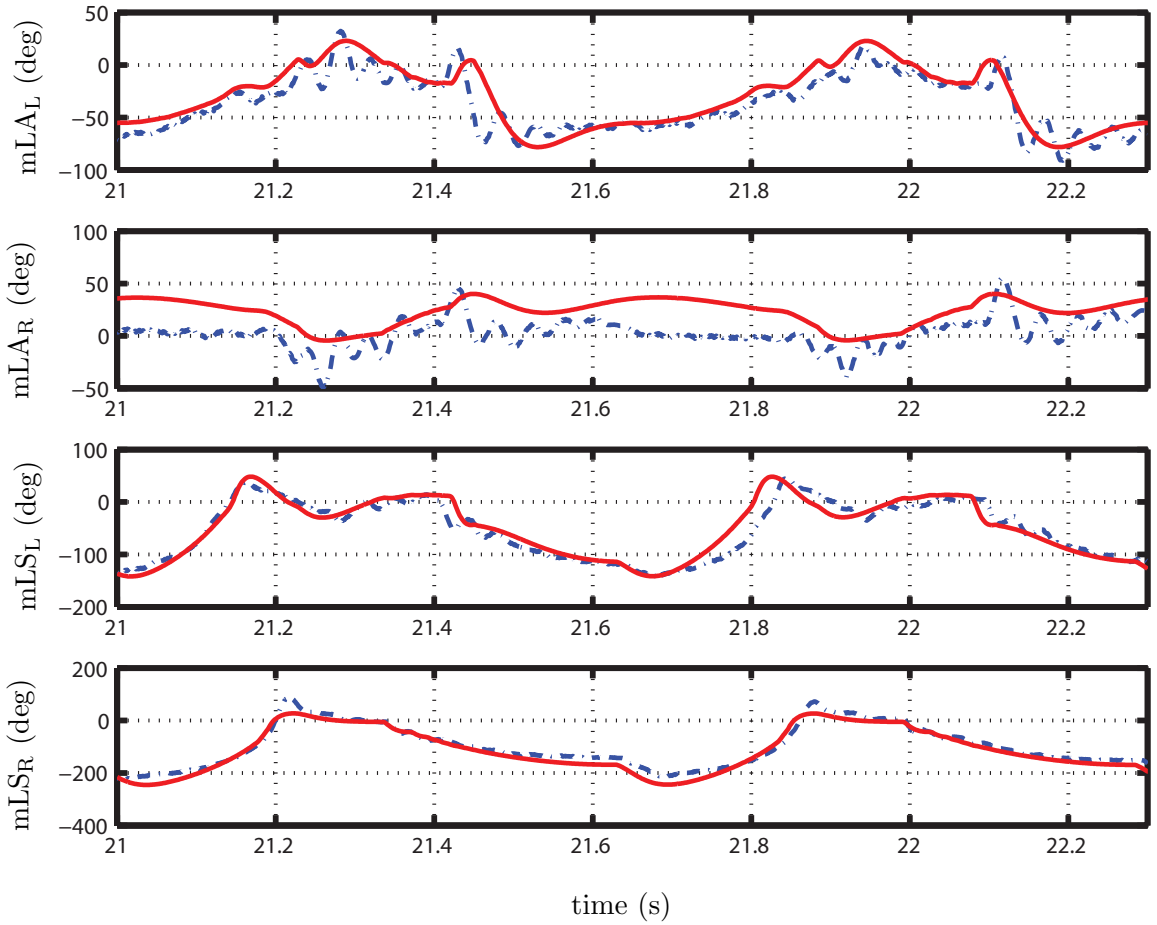


Figure 4.21: Validation data from the second hopping experiment. Cable stretch for the simulation is indicated by solid red lines, while cable stretch for the experiment is indicated by dotted blue lines. When the gear ratios are taken into account, 100 deg of cable stretch results in 4.25 deg of displacement in q_{LA} and 3.18 deg of displacement in q_{LS}.

for the 31st and 32nd hops. Figures 4.18 and 4.19 depict joint position angles. It is observed that the period of the experimental data is longer than that of the simulation results by approximately 30 ms. Because the hopping controller regulates four outputs computed from q_{T}, $q_{\mathrm{LA_L}}$, $q_{\mathrm{LA_R}}$, $q_{\mathrm{mLS_L}}$, and $q_{\mathrm{mLS_R}}$, it could be argued that the closeness of the simulated and experimental values is a reflection of the controller. However, the spring compressions $q_{\mathrm{Bsp_L}}$ and $q_{\mathrm{Bsp_R}}$, as well as the horizontal and vertical hip positions p^h_{hip} and p^v_{hip}, are unregulated, and Figure 4.19 shows that these variables are captured by the model.

Figure 4.20 depicts joint torques. The simulation predicts the joint torques observed in the experiment with an RMS error of 1.0 N-m. The ability to predict torque is an accepted measure of fit for models of mechanical systems [105, 104, 79, 111, 60].

Figure 4.21 compares the measured and simulated cable stretch, in degrees of motor rotation. Up to 200 deg of cable stretch is observed. The maximum error in the modeled response occurs in the right leg angle. The predicted ground reaction forces are not compared to experimental data because MABEL is not equipped with force sensors.

4.8 Conclusions

Parameter identification of MABEL, a 5-link bipedal robot with a compliant transmission, has been studied. For each side of the robot, the transmission is composed of three cable differentials that connect two motors to the hip and knee joints in such a way that one motor controls the angle of the virtual leg consisting of the line connecting the hip to the toe, while the length of the virtual leg is controlled by a second motor connected in series with a spring. The springs serve both to isolate the rotor inertia of the leg-shape motors from the impact forces at leg touchdown and to subsequently store energy when the support leg must decelerate the downward motion of the robot's center of mass.

The robot is equipped with 14 encoders to measure motor, pulley, and joint angles, as well as contact switches at the ends of the legs to detect impact with the ground. Neither force sensors, torque sensors, nor accelerometers are available. To circumvent these limitations, the identification procedure took advantage of the modular nature of the robot. By selectively disconnecting cables in the transmission, various elements can be isolated for

individual study, simplifying the parameter identification process.

MABEL has two kinds of compliance. One is the unilateral, fiberglass springs designed into the transmission. The other source of compliance is unplanned and arises from the stretching of the cables between the pulleys. The compliance of the unilateral spring was obtained by means of static loading experiments. The compliance from cable stretch was estimated from short-duration hopping experiments. From the same data set, the parameters for a compliant ground model were also roughly estimated. The experiments lasted only a few hops because we were unable to stabilize the hopping gait without a model; trial and error in the laboratory did not work.

The utility of the model was illustrated by using it to redesign the hopping controller. When the redesigned controller was implemented on the robot, 92 hops were made, at which point the experiment was terminated. The utility of the model will be further illustrated in subsequent chapters.

CHAPTER V

Control Design for Accommodating Large Step-down Disturbances

In this chapter, we propose a new control policy which allows MABEL to step off platforms that are at 20 cm high, without falling. The robot is provided information on neither where the step down occurs, nor by how much.

The remainder of the chapter is organized as follows. Section 5.1 provides the design of a baseline controller reported in [100] and an initial step-down experiment. A switching controller is designed in Section 5.2 and the proposed controller is verified on a detailed simulation model in Section 5.3. Experimental result of the new controller are provided in Section 5.4. Conclusions and future directions are given in Section 5.5.

5.1 Baseline Controller

5.1.1 Feedback Design

MABEL's baseline feedback controller is designed using the method of virtual constraints [116]. The particular controller used here has been reported in [100, 98]. The method of virtual constraints begins with the choice of outputs which depend on only configuration variables and take the form

$$y = h(q_{\mathrm{s}}) = h_0(q_{\mathrm{s}}) - h^d\left(s(q_{\mathrm{s}}), \alpha\right). \tag{5.1}$$

In the baseline controller, the *controlled variables* are

$$h_0(q_s) = \begin{bmatrix} q_{\mathrm{mLS_{st}}} \\ q_{\mathrm{LA_{sw}}} \\ q_{\mathrm{mLS_{sw}}} \\ q_{\mathrm{Tor}} \end{bmatrix} \tag{5.2}$$

and $h^d\left(s(q_s), \alpha\right)$ is a vector representing the *desired evolution* of the controlled variables as a function of $s(q_s)$, replacing time in a standard tracking controller. The function $s(q_s)$ is designed to be strictly monotonically increasing over the course of a step. If a feedback can be found such that the output y is driven asymptotically to zero, then the solutions of the closed-loop system asymptotically satisfy $h(q_s) = 0$, which has the form of a holonomic constraint on a mechanical system (for more information about virtual constraints, see [22]).

In the baseline controller, the desired evolution of the controlled variables in (5.2) is specified by the functions $h^d_{\mathrm{mLS_{st}}}$, $h^d_{\mathrm{LA_{sw}}}$, $h^d_{\mathrm{mLS_{sw}}}$, and h^d_{Tor}, respectively, and assembled as

$$h^d\left(s, \alpha\right) = \begin{bmatrix} h^d_{\mathrm{mLS_{st}}}\left(s, \alpha\right) \\ h^d_{\mathrm{LA_{sw}}}\left(s, \alpha\right) \\ h^d_{\mathrm{mLS_{sw}}}\left(s, \alpha\right) \\ h^d_{\mathrm{Tor}}\left(s, \alpha\right) \end{bmatrix}, \tag{5.3}$$

where α is a vector of real numbers parameterizing the virtual constraints. Furthermore, s is selected as

$$s(q_s) = \begin{cases} \frac{\theta(q_s) - \theta^+}{\theta^- - \theta^+}, & \theta^+ < \theta(q_s) < \theta^- \\ 1, & \theta(q_s) \geq \theta^- \\ 0, & \theta(q_s) \leq \theta^+ \end{cases} \tag{5.4}$$

where θ is the absolute angle formed by the virtual compliant leg relative to the ground, that is,

$$\theta(q_s) = \pi - q_{\mathrm{LA_{st}}} - q_{\mathrm{Tor}}, \tag{5.5}$$

and θ^+ and θ^- are the values of $\theta\left(q_s\right)$ at the beginning and end of a step, respectively.

How to construct the functions in $h^d(s,\alpha)$ from Bézier polynomials and how to choose the parameters to create a periodic walking gait in the closed-loop system are explained in [116] and [100]. The main idea is to select α to minimize a cost function representing energy supplied by the actuators, normalized by step length, with the minimization subject to boundary conditions that specify a periodic solution, actuator magnitude and power limitations, friction limits in the ground contact model, swing-leg clearance, and desired walking speed.

In principle, the virtual constraints can be implemented on the robot by any feedback capable of driving y to zero. In the experiments described below, we use the feedforward-plus-PID-controller,

$$u^{F-PD}(x_{\mathrm{s}}) = u^*\left(s(q_{\mathrm{s}}),\alpha\right) - K_P y - K_D \dot{y} - K_I \int y \mathrm{d}y, \tag{5.6}$$

where $u^*\left(s(q_{\mathrm{s}}),\alpha\right)$ is the nominal torque along the periodic orbit determined from the parameter-optimization problem when designing the virtual constraints, and y is defined in (5.1). The asymptotic stability of the periodic orbit under this feedback law is verified on the model with a Poincaré map, as explained in [116] and [100].

The above process results in the virtual constraints depicted in Figure 5.1. These constraints correspond to the nominal walking gait presented in [100], with average walking speed of approximately 1.0 m/s. Here, we modify the nominal virtual constraints so that the end of the swing leg at mid-stance can clear a 2 cm obstacle, allowing the robot to step onto a platform before stepping off it. Henceforth, we call this the *baseline controller*.

5.1.2 Baseline Step-down Performance

As reported in [77], using the control law (5.6) and the virtual constraints of Figure 5.1, MABEL can accommodate a 2.0 inch (5.08 cm) step-down disturbance. The experiment was conducted as follows. MABEL was put in motion, walking on an initially flat floor. At the end of each lap, MABEL walked up a stair-stepped ramp[1], and then stepped off

[1]The ramp and platform are constructed from sheets of plywood that are 0.5 inch and 1.0 inch thick. This explains the use of non-SI units. Figure 5.2 illustrates how the ramp and platform are constructed from a combination of 0.5 inch and 1.0 inch (2.54 cm) thick sheets of plywood.

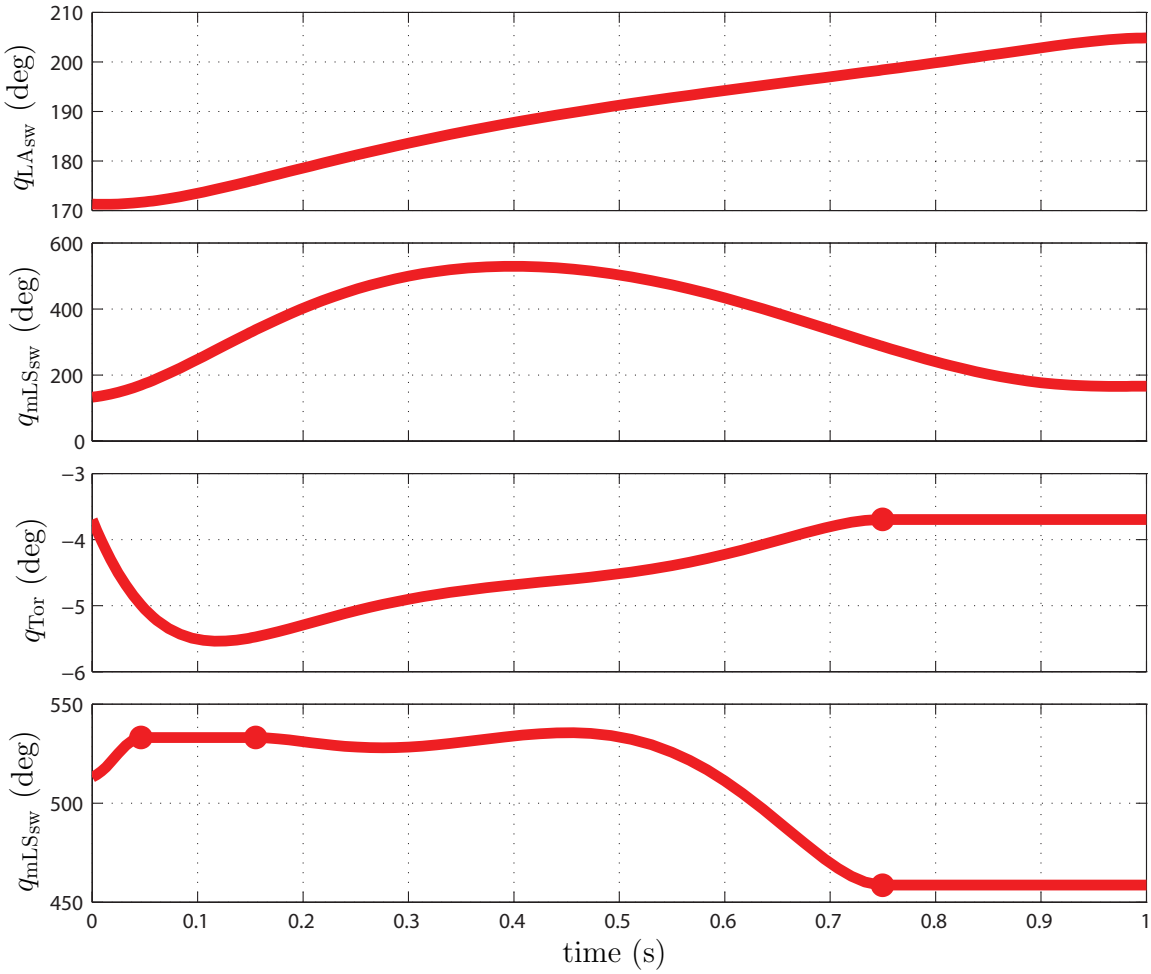


Figure 5.1: Virtual constraints of the baseline controller.

the platform. The height of the platform was increased by 0.5 inches each lap until the robot fell when the platform height was increased to 2.5 inch (6.35 cm). MABEL fell after stepping off the 2.5 inch platform because the leg broke on the ensuing step; the video is available at [73].

Figure 5.3 shows the torso angle data; it can be seen that the feedback system overreacts when correcting the forward-pitching motion of the torso, causing a second, very rapid, forward-pitching motion of the torso. Because the angle of the swing leg was controlled relative to the torso, the swing leg rotated forward rapidly as well and impacted the ground with sufficient force to break the leg. Though not reported in [77], the experiment was repeated several times, with the same result, namely, a broken leg following a 2.5 inch step-down.

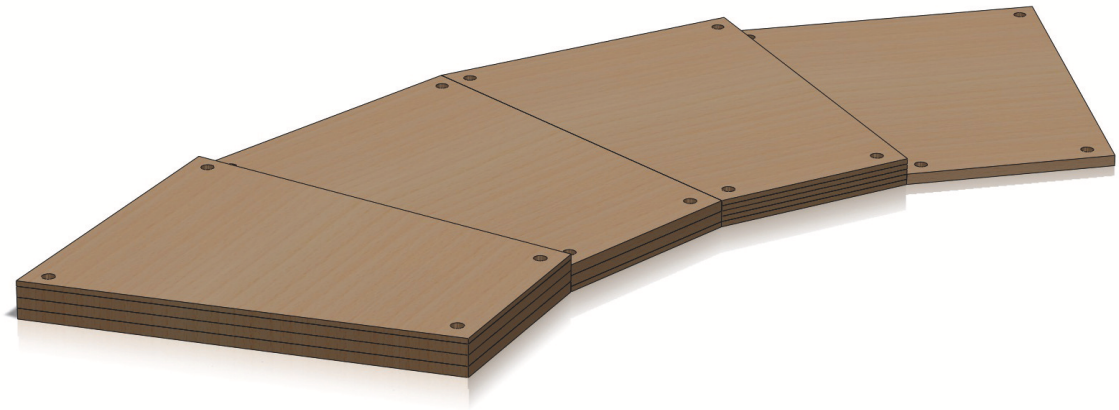


Figure 5.2: Plywood boards used to form a stair-stepped platform. By combining boards that are 1 inch (2.54 cm) and 0.5 inch (1.27 cm) thick, platforms of a desired height are assembled.

Further analysis of experimental data was carried out in [77] to study the impact forces that caused the leg to break. Using the impact model of [52], the contact intensity I_F at the leg end was estimated from the experimental data. I_F has units of $N \cdot s$ and represents, roughly speaking, the integral of the contact force over the duration of the contact event. Figure 5.4 shows the estimated contact intensity when walking on flat ground and when stepping off several raised platforms. The data indicate that, upon stepping off the 2.5 inch platform, the impact intensity of the second step 47 ($N \cdot s$) was more than three times as intense as the impact intensity 13 ($N \cdot s$) of walking on flat ground.

Work presented in [77] went on to show how to design a switching controller that resulted in MABEL stepping off a 3.5 inch (8.89 cm) platform without falling. The controller reported there used the same form of the virtual constraints shown in (5.1) and (5.3), with the virtual constraints of Figure 5.1 specially designed to reduce torso pitching. In the next section, a switching controller is proposed that will result in a dramatic increase in performance: MABEL stepping off an 8.0 inch (20.32 cm) platform without falling.

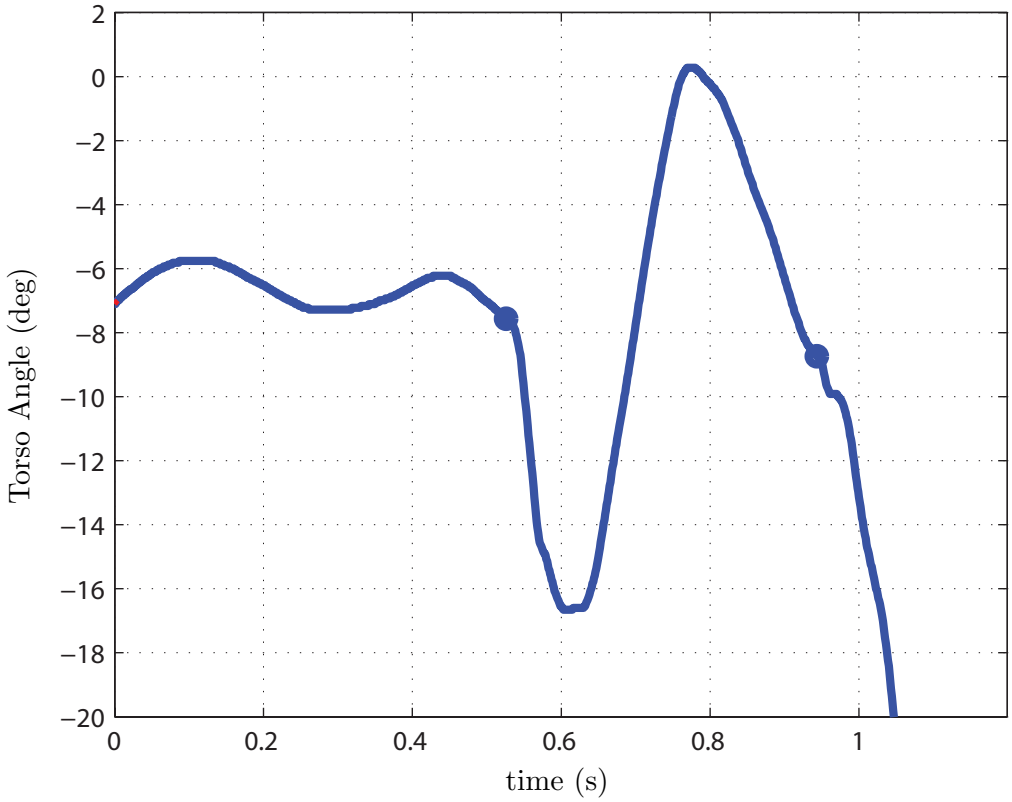


Figure 5.3: Experimental and data of the torso angle when stepping down from the 2.5 inch platform. The forward direction is counterclockwise, and hence corresponds to decreasing angles. The blue circles show when the swing leg impacts the ground. The torso undergoes a large oscillation as a result of the disturbance at step-down.

5.2 A Switching Controller Based on Virtual Compliance

5.2.1 Overview of Basic Controller

The baseline step-down experiments showed that MABEL fell on the second step of a large step-down, and not at the moment of step-down itself. The cause of the fall is the torso pitching forward too rapidly after leg impact with the ground. To attenuate the amount of torso pitching following a large step-down, we adopt the idea of a switching controller from [124].

The height of the platform, or equivalently, the depth of the step down, can be immediately computed at impact from the lengths of the robot's legs and the angles of its joints. If the calculated height of the platform is greater than a threshold scalar value ΔH which is

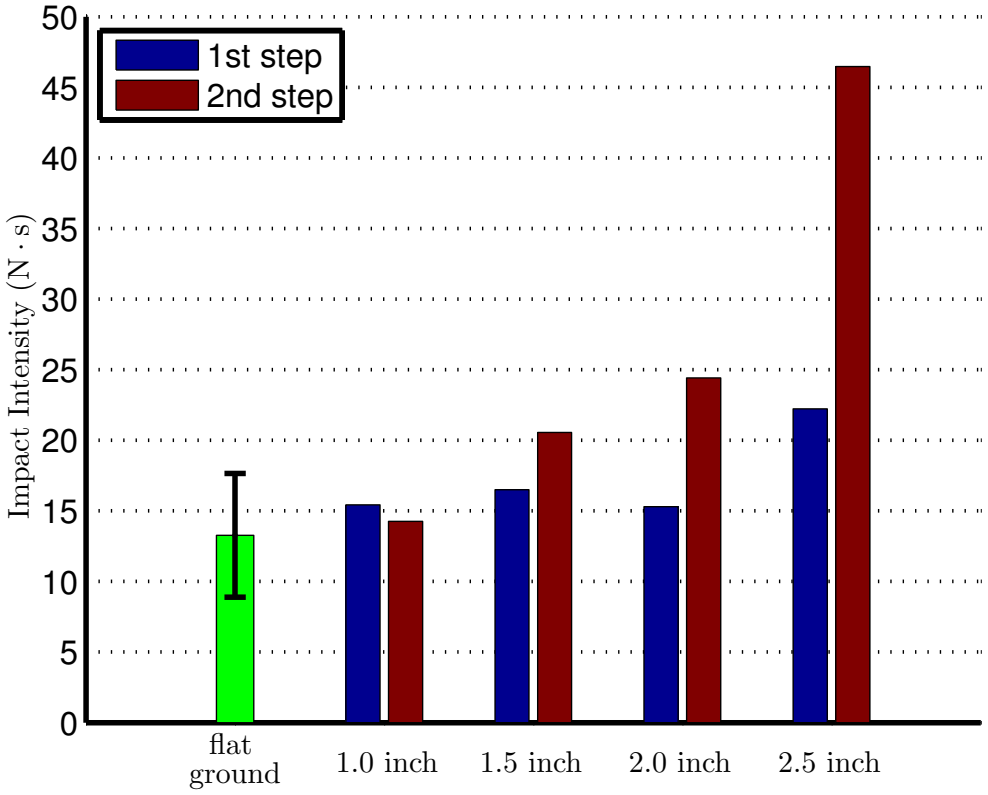


Figure 5.4: Impact intensity calculated from the baseline step-down experiments. Only the vertical component is displayed. The green bar shows the average impact intensity when walking on flat ground; the bars show ± one standard deviation. The blue bar shows the impact intensity upon step-down, while the red bar shows impact intensity for the ensuing step. The larger impact intensity on the step following the 2.5 inch (6.35 cm) step-down leads to a mechanical fuse activating in the shin, which separates the leg into two pieces.

being set to 3 cm in this Chapter, the baseline controller is replaced for one step with a controller whose purpose is to attenuate pitching of the torso from the step-down disturbance. Then, at the beginning of the very next step, the baseline controller is re-applied.

5.2.2 Active Force Control

The new controller, called a 'active force control', imposes virtual holonomic constraints on only three variables, $q_{\mathrm{LA_{sw}}}, q_{\mathrm{mLS_{sw}}}$, and q_{T}, instead of four variables, as in the baseline controller. In particular, the system input corresponding to the stance motor leg shape is left free and not used for imposing a virtual constraint. Recall that this motor is in series with a physical spring in the drivetrain, as shown in Figure 3.1b. Following an idea

developed in [98, 99] for bipedal running on MABEL, we use the torque input of this motor to create an additional *virtual compliant element* by defining the feedback,

$$u_{\mathrm{mLS_{st}}}(x) = -k_{\mathrm{vc}}\left(q_{\mathrm{mLS_{st}}} - q_{\mathrm{mLS_{vr}}}\right) - k_{\mathrm{vd}}\left(\dot{q}_{\mathrm{mLS_{st}}}\right). \tag{5.7}$$

This feedback essentially turns the leg shape motor into a *shock absorber* with stiffness k_{vc}, damping k_{vd}, and rest position $q_{\mathrm{mLS_{vr}}}$. As will be seen, this method of creating a virtual compliant element serves to maintain good ground contact forces (friction cone is respected and normal component is positive) during large step-down experiments.

In principle, each of the virtual constraints for the three controlled variables $q_{\mathrm{LA_{sw}}}$, $q_{\mathrm{mLS_{sw}}}$, and q_{T} can be redesigned in the active force control. We found that the baseline virtual constraints could be retained for the swing leg angle and shape; it was only necessary to redesign the virtual constraint for the torso. When redesigning the virtual constraint for the torso, the first and last coefficients of the Bézier polynomials from the baseline controller are retained; the coefficients between them, denoted hereafter by α_{T}, will be selected through optimization.

The design parameters that are to be chosen through optimization are grouped into a vector denoted

$$\Theta^{\mathrm{SD}} := \left[\alpha_{\mathrm{T}}, k_{\mathrm{vc}}, k_{\mathrm{vd}}, q_{\mathrm{mLS_{vr}}}\right]^T. \tag{5.8}$$

In general, as in [115], a suite of parameters could be pre-computed for a discrete set of platform heights, such as {5 cm, 10 cm, 15 cm, 20 cm}, and the controller would then select an appropriate parameter vector, or the baseline controller, based on the estimate of platform height made at step-down. We found, however, that when the virtual compliant element in (5.7) is used, a step-down controller designed for 20 cm can accommodate step-downs of 3 cm to 20 cm, and the baseline controller can accommodate variations in ground height from a 2 cm rise to a 3 cm step-down, without the controller being provided *a priori* information on the amount of ground height variation. This finding obviously leads to a very simple switching policy.

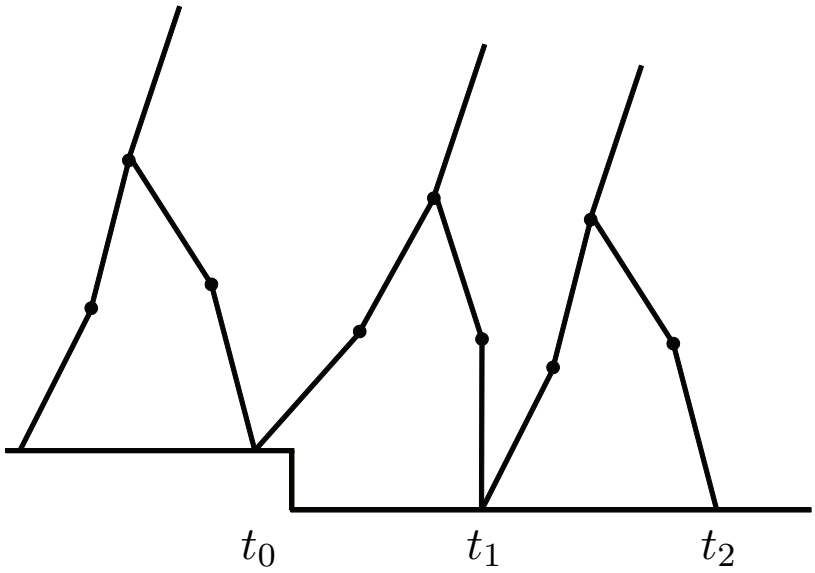


Figure 5.5: Design of step-down controller. Time t_0 corresponds to the end of the last step on the platform; t_1 is the end of the step-down; and t_2 is the end of the ensuing step.

We next describe how the parameter vector in (5.8) is selected via optimization.

5.2.3 Optimization Process for Step-down Controller

The optimization process is based on the simplified design model introduced in Section 3.1, with the switching surface in (3.10), modified as in (3.12) to account for a change in ground height at impact.

Assume the robot is on the periodic orbit corresponding to the baseline controller, and hence is walking on flat ground. As in Figure 5.5, let t_0 be time that step-down is initiated, let t_1 be the time the swing leg impacts the ground, and let t_2 be the end of the next step, assuming it occurs. The robot is operating under the baseline controller over the interval $[t_0, t_1)$, and under the step-down controller over the interval $[t_1, t_2]$. Once the step-down height H is specified, $x_{\mathrm{s}}(t_1)$, the state of the robot at time t_1, is known. A numerical optimization problem is posed so that the trajectory under the step-down controller can be continued in such a way that the robot will not fall. In principle, the optimization could consider several steps, but only one step is considered here.

Objective: Select Θ in (5.8) to minimize peak-to-peak amplitude of torso oscillation as

defined by,

$$J_{\Theta^{\text{SD}}} := \max_{t\in[t_1,t_2]}\{q_{\text{T}}(t)\} - \min_{t\in[t_1,t_2]}\{q_{\text{T}}(t)\}, \tag{5.9}$$

where $q_{\text{T}}(t)$ is trajectory of the torso angle. The cost function J_Θ is optimized subject to the following constraints:

(a) positive horizontal swing toe position at the end of the step,

$$p_2^h(t_2) > 0; \tag{5.10}$$

(b) bound on the ratio of tangential to normal ground reaction forces experienced by the stance leg end,

$$\max_{t\in[t_1,t_2]} \left\{ \frac{F_1^T(t)}{F_1^N(t)} \right\} < \mu_s; \tag{5.11}$$

(c) minimum normal ground reaction force experienced by the stance leg end,

$$\min_{t\in[t_1,t_2]}\{F_1^N(t)\} > C, \quad \text{for some } C > 0; \tag{5.12}$$

(d) avoid premature impact,

$$s(t_2) > 1 - \delta, \text{ where } \delta \ll 1 \tag{5.13}$$

(e) upper bound on the magnitude of torso angular velocity

$$\max_t \{|\dot{q}_{\text{T}}(t)|\} < \gamma_{\dot{q}_{\text{T}}}, \text{ for some } \gamma_{\dot{q}_{\text{T}}} > 0. \tag{5.14}$$

The optimization is conducted over a transient phase of the gait, and thus constraints are not required to impose periodicity of an orbit. We have observed that solutions to the above optimization problem tend to steer the robot so that the configuration variables at time t_2 are near their values on the periodic orbit. We conjecture that this is because the

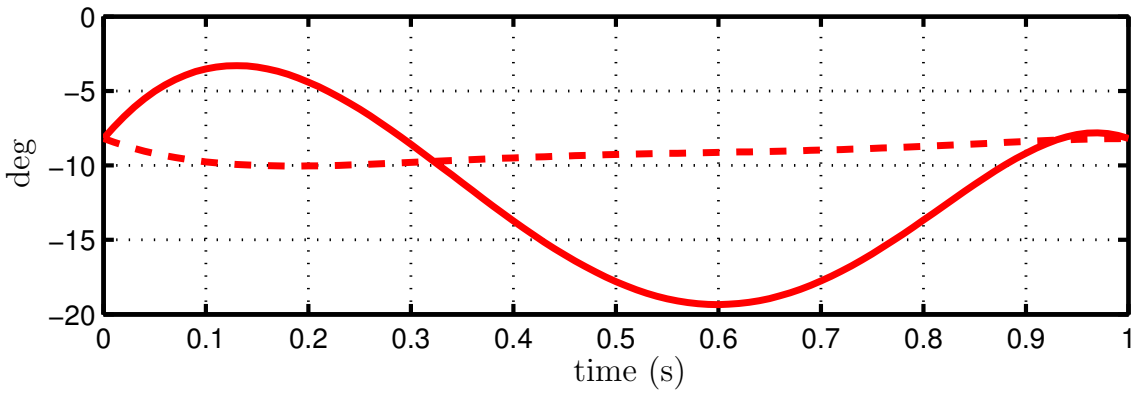


Figure 5.6: Torso virtual constraints. Solid red line minimizes peak-to-peak amplitude of torso oscillation after a 20 cm step-down, while the dashed red line is from the baseline controller.

two virtual constraints for the swing leg, as well as the ending value for the torso virtual constraint, are inherited from the baseline periodic orbit.

MATLAB's constrained optimization routine `fmincon` is used to perform the numerical search outlined above. Setting $H = -0.2$ m, corresponding to a 20 cm step-down, $\gamma_{\dot{q}_T} = 250$ deg/sec, $C = 200$ N, $\delta = 0.05$, and $\mu_s = 0.5$, yields the parameter vector

$$\Theta^{\text{SD}} = [0.1417, \quad -0.2338, \quad -0.7765, \quad -0.0529, \quad 2.1088, 0.1297, \quad 2.7028]^T. \qquad (5.15)$$

Figure 5.6 shows the torso virtual constraint resulting from the optimization.

The resulting switching controller is then applied to the simplified design model with a 20 cm step-down. The torso trajectory is shown in Figure 5.7. It is observed that the torso oscillates approximately 11 deg during the step following the step-down (red solid line) and then quickly converges to its nominal trajectory. On the other hand, under the baseline controller (red dashed line), the torso noticeably overshoots when returning to the nominal lean angle. Figure 5.8 shows the normal ground reaction force and pulley B_{spring} angle, which are important indicators of ground contact condition.

Before implementing the controller on the robot, it is evaluated on the detailed model explained in Chapter III. As mentioned in Chapter III, the validation model takes into account more aspects of the physical robot, but it is too unwieldy for controller design when optimization is involved.

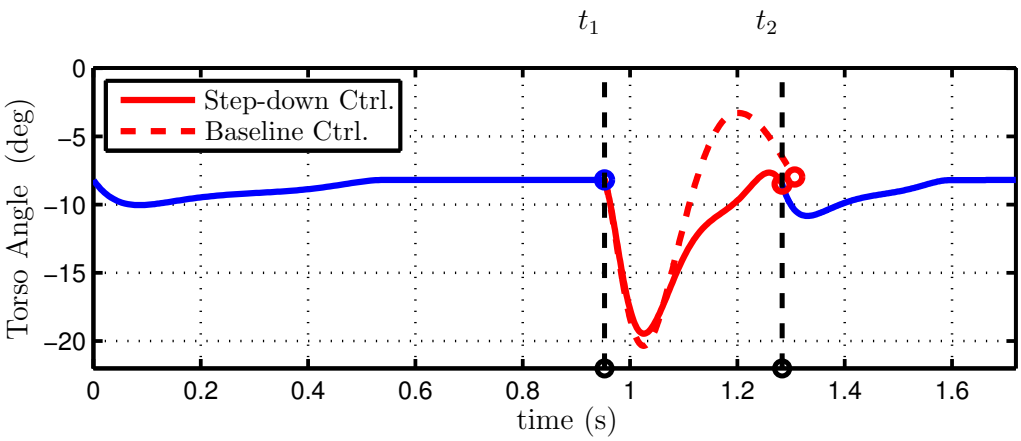


Figure 5.7: Simulated torso trajectory with optimal parameter vector Θ^{SD} on the simplified design model for a step-down height of 20 cm. The switching controller is shown with a solid line, while a dashed line represents the baseline controller. Red line indicates the step following step-down.

5.3 Controller Verification on Detailed Model

As explained in Chapter 3.2, the model used for control design does not fully reflect experimental reality due to the following reasons: cable stretch in the robot's drivetrain; asymmetry due to the boom radius not being large enough; the simplified impact model assumes an instantaneous double support phase, whereas, in experiments, the double support phase lasts approximately 20 ms. By including cable stretch as a spring-damper, the boom dynamics to account for asymmetry side-to-side, and a ground model comprised of compliant ground and LuGre friction model [13, 80] in the mathematical model, the accuracy of the model is significantly improved. However, because of the complexity of this model, simulations of the *detailed model* take 20 times longer than the *simplified design model*. Hence, this model is not appropriate for optimization processes which may require thousands of simulations.

Therefore, to take advantage of each model's strengths, low computational effort for the simple-design model and high accuracy for the detailed model, iterative controller design is conducted on the simple model first, and then the designed controller is tested on the detailed model, before implementing it on the robot.

As part of implementing the proposed controller on the detailed model, two modifications are made to account for the cable stretch which is the most critical reason for model

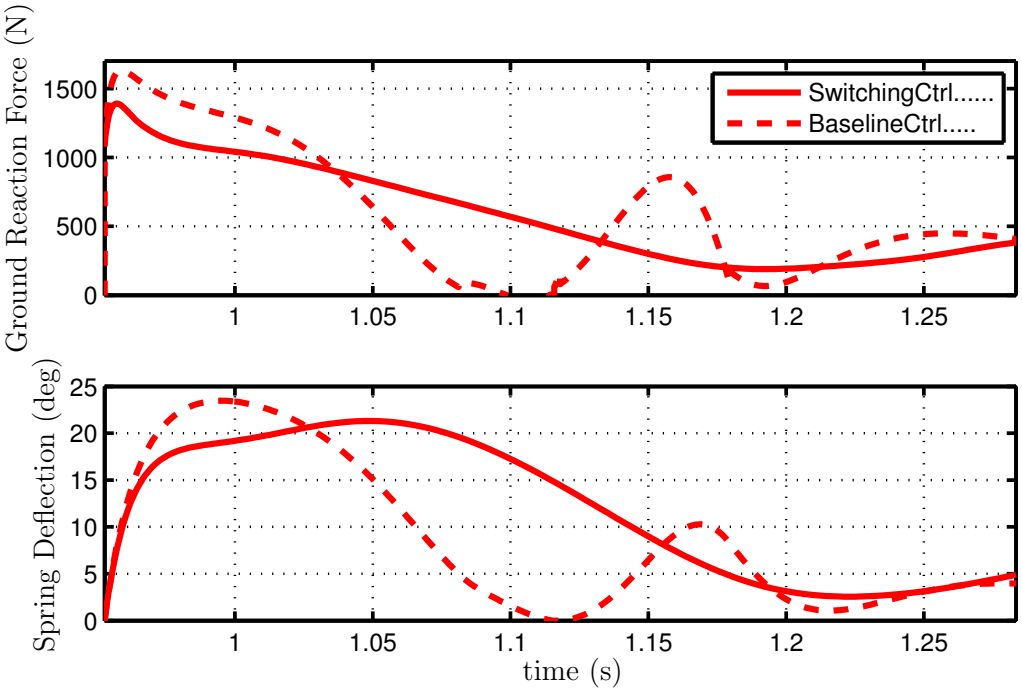


Figure 5.8: Simulation with optimal parameter vector Θ^{SD} on the simplified design model for a step-down height of 20 cm. The switching controller is shown with a solid line, while a dashed line represents the baseline controller.

discrepancy: (A) the coefficients of the virtual compliance in (5.7) are modified so that the series connection of the compliance due to the cable stretch and the virtual compliance has the effective compliance specified by the optimization process (see [98] and Appendix E for detail); (B) the swing leg height is increased according to the platform height calculated at impact to prevent foot scuffing; (C) in addition, a one-step transition phase is added between the step-down and baseline controller in order to connect the two phases smoothly. With these modifications to the proposed controller, the simulation results in MABEL successfully stepping off a 20 cm platform.

Figure 5.9 shows q_{T} and $q_{\mathrm{Bsp_{st}}}$. Despite the significant torso oscillation after step-down, the calculated impact intensity 20.1 N $\cdot$ s from the 20 cm step-down simulation data is less than half of the impact intensity 46.5 N $\cdot$ s observed in the 6.35 cm step-down under the baseline controller (see Figure 5.4). Furthermore, it is observed that the torso oscillation damps out rapidly over the ensuing steps and the vertical ground reaction force is positive.

5.4 Experiment

The switching controller is now evaluated on the robot. In the experiment, MABEL starts walking on a flat floor, walks up a ramp to a 17.78 cm platform and steps off, and completes a second lap before being stopped by a researcher. With the robot stopped, the platform is increased to 20.32 cm. The robot was then put back in motion, walked 19 steps, and stepped off the platform without falling.

Experimental data from the 20.32 cm step-down are shown in Figures 5.10 and 5.11. Following the step-down, the amplitude of torso oscillation is approximately 15 deg, while the detailed model predicted a value of 13.5 deg. Remarkably, with the switching controller, the torso oscillates less after the 20.32 cm step-down than it does after the 2.5 inch (5.08 cm) step-down test when using the baseline controller, even though the platform is more than three times as high. In addition, the calculated impact intensity at the second step is 23.8 $\mathrm{N} \cdot \mathrm{s}$, which is also smaller than the value of the 2.5 inch (5.08 cm) step-down test.

The spring deflection peaks at slightly over 40 deg just after step-down and reaches zero for a short interval of time at the end of the step, as shown in Figure 5.10. Two steps after the step-down event, the torso and and spring deflection are indistinguishable from steady-state walking on flat ground, showing that the gait induced by the switching controller is smoothly steered to the gait of the baseline controller.

Snapshots from video capture and a stick figure illustration of the experiment are shown in Figure 5.11. This data also demonstrate that the walking gait converges quickly to its steady-state characteristics following the step-down event. The video is available at [74].

5.5 Conclusion

A switching controller has been designed to handle blind step-downs of considerable height. Experimentation with this controller showed MABEL stepping off a 20 cm platform. The next chapter will focus on stepping onto a platform without prior knowledge of its location or height.

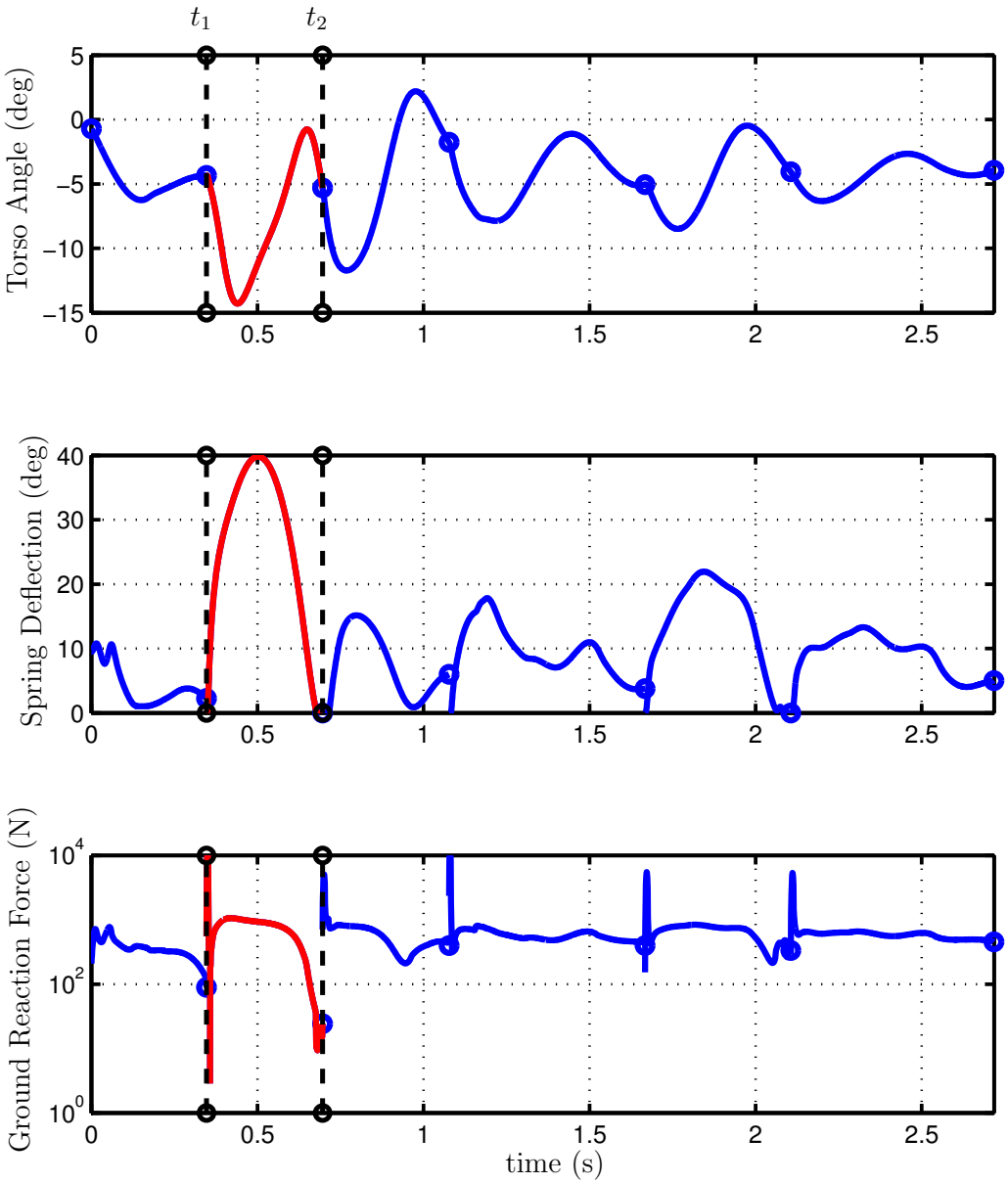


Figure 5.9: Step-down Simulation data from the detailed model. Solid red line represents the step following step-off.

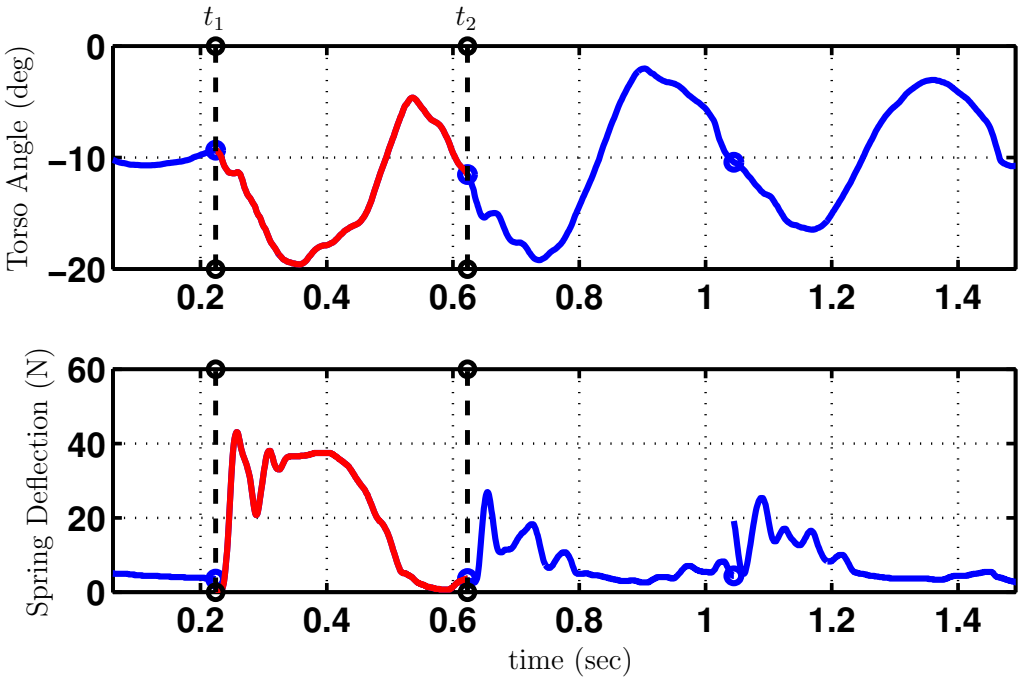
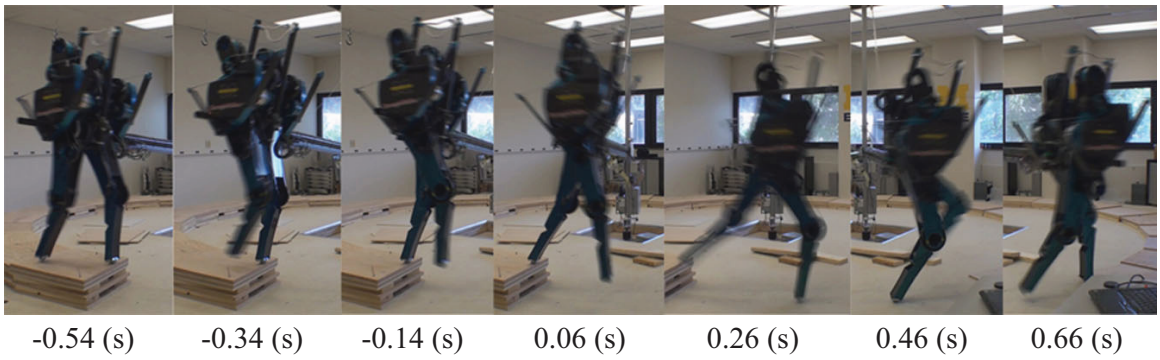
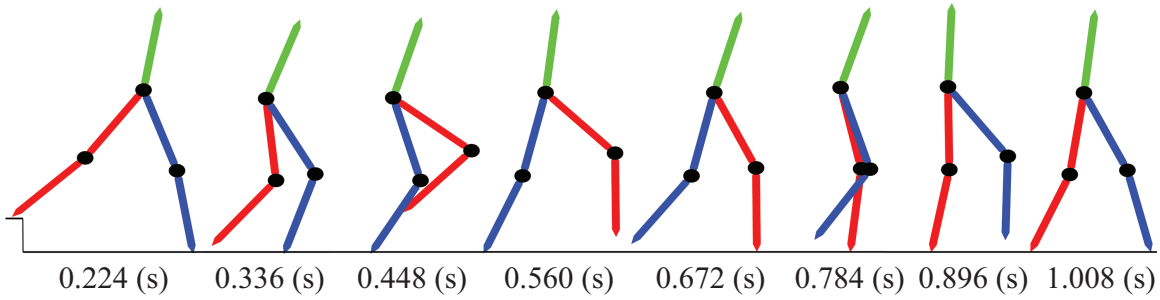


Figure 5.10: Experimental data for the the 20.32 cm step-down. The step-down occurs at 0.224 (s). The following step is marked as a red line.



(a)



(b)

Figure 5.11: Experimental data for the 20.32 cm step-down: snapshot and stick-figure. Snapshots from video capture are shown in (a), while (b) shows a stick-figure illustration of the two steps following the step-down disturbance. The video is available at [74].

CHAPTER VI

Control Design for Accommodating Large Step-up Disturbances and for Tripping Recovery

As revealed in the successful 20 cm step-down experiments in Chapter V, by activating the specifically designed step-down controller in response to a step-down disturbance, MABEL could not only keep the vertical ground reaction force positive, causing the stance toe to stay in contact with the ground, but also reduce torso oscillation by a significant amount, thereby attenuating the impact force on the swing toe at the end of the ensuing step of a step-down event.

Motivated by this experimental result, in this chapter, we seek a controller for accommodating large step-up disturbance, using *active force control* as explained in Chapter V, which replaces stance knee control with a virtual compliant element. A similar optimization process is posed to select parameters of the virtual compliant element for accommodating step-up disturbances: stiffness k_{vc}, damping k_{vd}, and rest position $q_{\mathrm{mLS_{vr}}}$. The control design will result in MABEL waking up of stairs with considerable height without falling.

In addition, a reflexive strategy to take care of tripping over obstacles is also sought in this chapter. As discussed in Chapter I, when robot is walking over unknown ground consisting of vertical steps, the robot's swing leg may strike vertical stairs or have a premature impact with the ground, unexpectedly, and eventually fall. Therefore, a reflexive strategy is required to respond and recover from tripping if the robot is to accommodate terrains with vertical obstacles without a priori information.

We look now for inspiration from biomechanical studies. It is shown in [24, 25] that

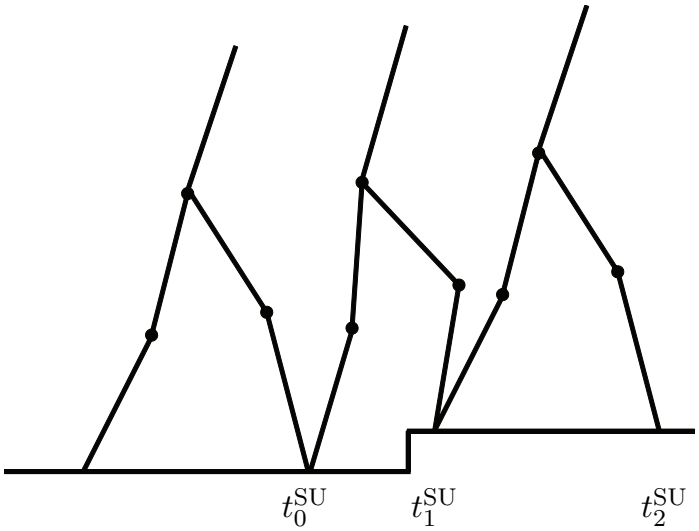


Figure 6.1: Design of step-up controller. Time t_0^{SU} corresponds to the end of the last step on the level ground; t_1^{SU} is the start of the step-up; and t_2^{SU} is the end of the step-up.

humans accommodate a tripping disturbance by changing their gait strategy. Cutaneous receptors are stimulated to detect when the trajectory of the swing foot during walking is obstructed, and a response which ensures the removal of the limb from the obstruction, as well as the safe continuation of the locomotor pattern is activated to deal with the detected obstacle.

In MABEL, this idea of a reflex strategy in response to tripping perturbations can be implemented in the context of a switching-control design. Tripping over obstacles can be detected by using the signal of the switch located in front of the shin and at the end of toe, replacing cutaneous receptors in humans.

6.1 Design of Step-up Controller

The step-up controller is activated in response to an impact with an obstacle that is higher than a threshold scalar value ΔH, which is set to 5 cm in this chapter. In the design of the step-up controller, we found that the baseline virtual constraints for the torso could be retained; only the virtual compliance and the virtual constraints for the swing leg need to be redesigned.

Because the legs are relatively light, the effects of swing leg movement on the robot's overall dynamics is far less dominant than the effects of the torso and stance leg. For this

reason, only the parameters in the stance leg's virtual compliance, namely stiffness k_{vc}, damping k_{vd}, and rest position $q_{\mathrm{mLS_{vr}}}$, are obtained by solving a constrained optimization problem. After the parameters of the virtual compliance are obtained, the virtual constraints for the swing leg shape and angle are designed by hand. In particular, the virtual constraints for the swing leg are designed so that knee is bent rapidly by repositioning the swing motor leg shape, and the swing leg is also moved backward at the beginning of the gait by repositioning the swing leg angle to avoids re-striking the obstacle during step-up. Subsequently, both the optimized virtual compliance parameters and the hand-tuned virtual constraints for the swing leg shape and angle are verified to satisfy the constraints of the constrained optimization problem with the optimization cost not being changed significantly.

The optimization parameters are assembled as,

$$\Theta^{\mathrm{SU}} := [k_{\mathrm{vc}}, k_{\mathrm{vd}}, q_{\mathrm{mLS_{vr}}}]^T, \tag{6.1}$$

and the optimization problem for the selection of Θ is posed as follows. Assume the robot is on the periodic orbit corresponding to the baseline controller, and hence is walking on flat ground. As displayed in Figure 6.1, let t_0^{SU} be the time that the robot is at the end of the last step on flat ground, let t_1^{SU} be the time step-up starts, and let t_2^{SU} be the end of the step-up. The robot is operating under the baseline controller with the modification on the virtual constraints for the swing leg over the interval $[t_0^{\mathrm{SU}}, t_1^{\mathrm{SU}})$. In this modified baseline controller, the virtual constraints for the swing leg are modified so that robot can clear a 10 cm obstacle. Over the interval $[t_1^{\mathrm{SU}}, t_2^{\mathrm{SU}}]$, the robot is operating under the step-up controller. Once the step-up height H is specified, $x_{\mathrm{s}}(t_1^{\mathrm{SU}})$, the state of the robot at time t_1^{SU}, is known. A numerical optimization problem is posed so that the trajectory under the step-up controller can be continued in such a way that the robot will not fall. While retaining the same objective function and constraints as in (5.9), (5.10), (5.10), (5.11), (5.12), (5.13), and (5.14), one additional constraint is added in this optimization problem,

$$q_{\mathrm{LS_{st}}}(t_2^{\mathrm{SU}}) < 5 \text{ deg}. \tag{6.2}$$

This constraint enforces that the robot ends the step-up with the stance leg shape nearly straight, which avoids starting the next step with a crouched configuration. Setting $H =$ 0.1 m, corresponding to a 10 cm step-up, $\delta = 0.07$, $\gamma_{\dot{q}_T} = 180$ deg/sec, $C = 200$ N, and $\mu_s = 0.5$, yields the parameter vector

$$\Theta^{\mathrm{SU}} = [1.2198, \quad 0.1417, \quad 2.7027]^T. \tag{6.3}$$

Hand-tuned virtual constraints for the swing leg angle $\alpha_{\mathrm{LA_{sw}}}$, and swing leg shape $\alpha_{\mathrm{mLS_{sw}}}$ are given by,

$$\begin{aligned} \alpha_{\mathrm{LA_{sw}}} &= [2.9894, 2.4625, 3.2327, 3.2277, 3.5248, 3.4748] \\ \alpha_{\mathrm{mLS_{sw}}} &= [2.3232, 6.3703, 30.0000, 30.0000, 15.0000, 3.8979] \end{aligned} \tag{6.4}$$

A simulation using the simplified design model is then performed for both the step-up controller with the optimized Θ^{SU} and the baseline controller. The results are shown in Figure 6.2. There is no notable difference in the torso angle between the baseline controller and step-up controller because the virtual constraints for the torso were identical. However, ground reaction force and spring deflection are improved by the step-up controller while the baseline controller fails to control those values to remain positive. Figure 6.3 shows the position of the swing foot throughout the step. The gray colored region represents the vertical obstacle. We can observe that swing foot does not touch the vertical step during step-up.

6.2 Tripping-reflex controller

In this section, a tripping-reflex controller (TR) is designed. Tripping occurs when the robot's swing foot experiences unexpected impacts. The impacts can arise due to obstacles, as shown in Figure 6.4a and 6.4c, or due to inadequate foot height, as shown in Figure 6.4b. The required response varies with the nature of the impact and it is thus important that the robot has adequate sensing to differentiate among the cases shown in Figure 6.4. For this purpose, MABEL has contact switches on the front of each lower shin, as shown in

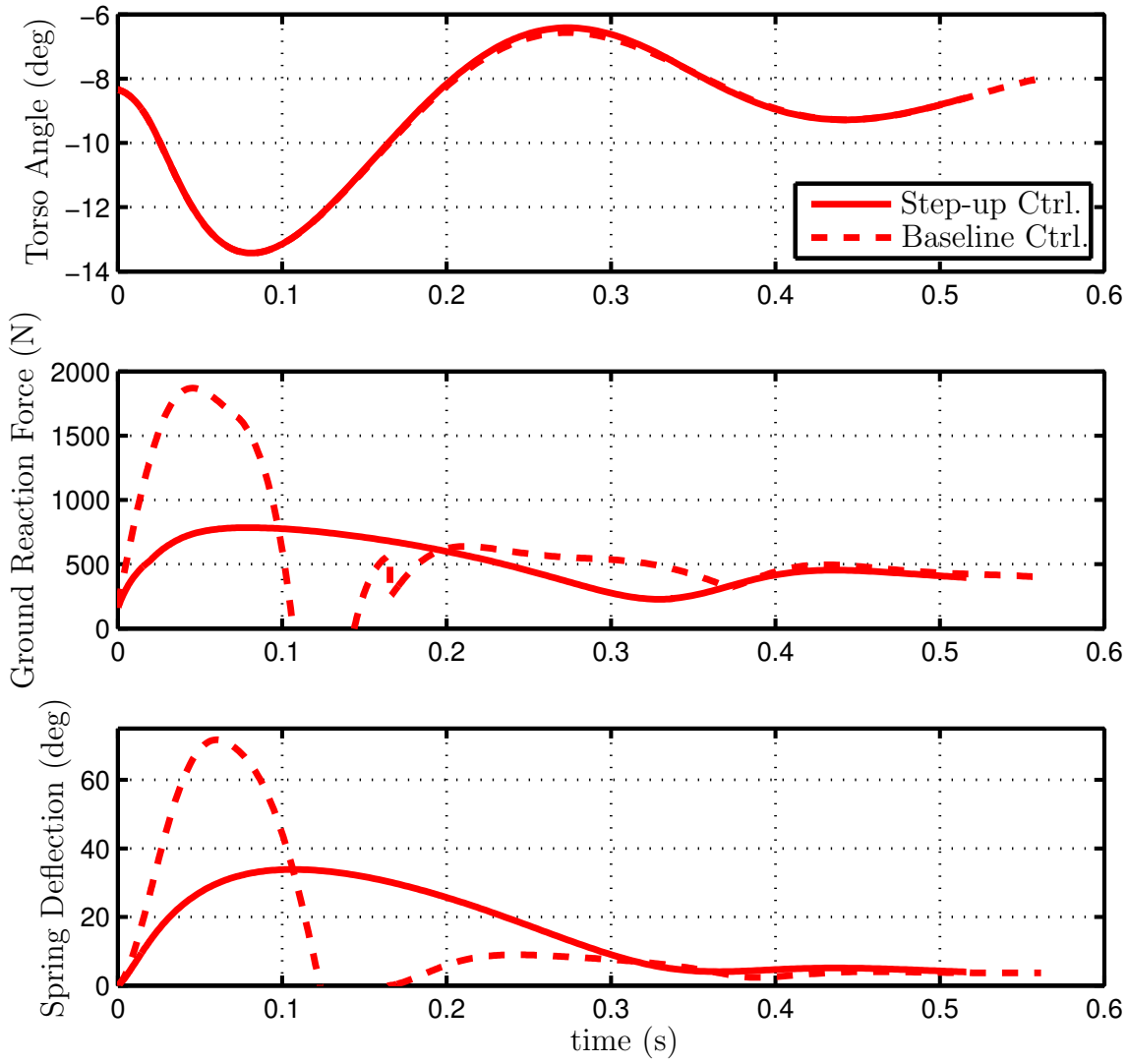


Figure 6.2: Simulation data with optimal parameter vector Θ^{SU} from the simplified design model for a step-up height of 10 cm. The step-up controller is shown with a solid line, while a dashed line represents the baseline controller.

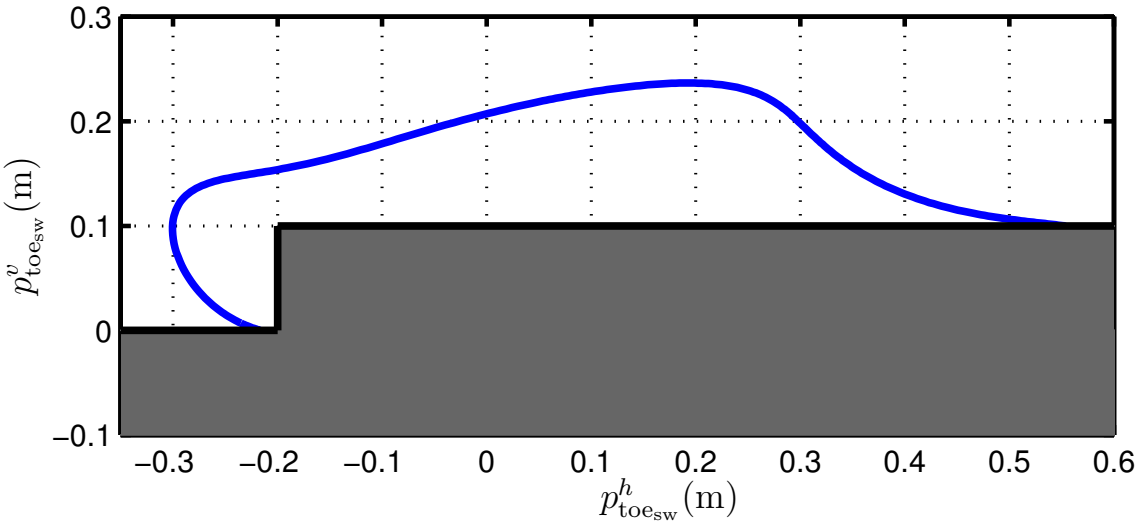


Figure 6.3: Horizontal position $p^h_{\text{toe}_{\text{sw}}}$ and vertical position $p^v_{\text{toe}_{\text{sw}}}$ of swing foot throughout the step. Grey colored region illustrates the vertical obstacle.

Figure 2.8, in addition to the traditional contact switches on the end of each leg.

It is suggested that humans adopt two different strategies for recovery from tripping, depending on whether it occurred in the early or late swing phase [24, 25]. A strategy of rapidly elevating the swing limb is activated in response to early tripping while a lowering strategy is used in response to late tripping. Similar strategies are applied here.

6.2.1 Reflexive Strategy for Late Tripping

Late tripping can be defined mathematically with respect to the robot's state and contact switch signals. If the frontal surface of the swing foot hits the obstacle ($sw_{\text{sh}} = 1$) with the step being close to the end ($s(q_{\text{s}}) \geq s_{\text{early}}$) as well as the absolute swing leg angle being advanced sufficiently ($q^{abs}_{\text{LA}_{\text{sw}}} := q_{\text{LA}_{\text{sw}}} + q_{\text{T}} \geq q_{\text{early}}$) to yield the next step starting with a value of $s(q_{\text{s}})$ close to 0, then tripping is classified as late tripping. Mathematically, it is equivalent that the robot's state enters the set which is given by,

$$\mathcal{S}_{\text{RL}} := \left\{ x_{\text{s}} \in T\mathcal{Q}_{\text{s}} \,\middle|\, s(q_{\text{s}}) \geq s_{\text{early}}, q^{abs}_{\text{LA}_{\text{sw}}} \geq q_{\text{early}}, sw_{\text{sh}} = 1 \right\}. \tag{6.5}$$

In response to late tripping, the rapid-lowering (RL) strategy of the swing leg is applied. The controller is not changed until the swing foot touches the ground. Because all of the

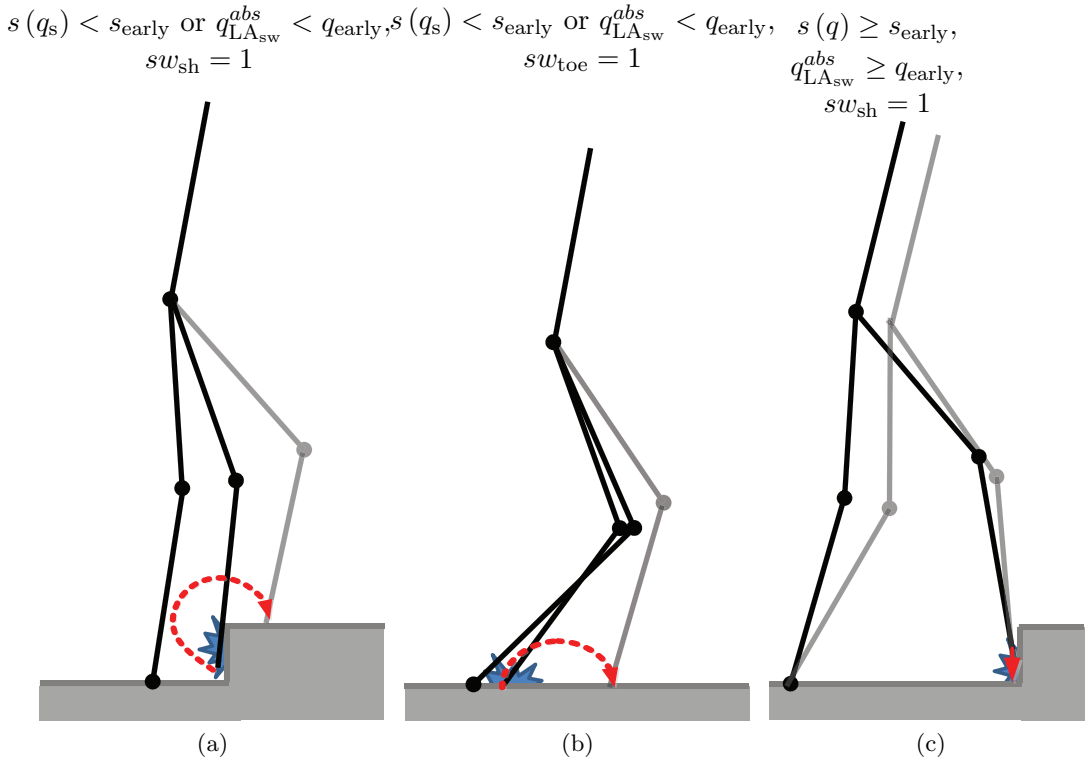


Figure 6.4: Various situations of tripping. The red line with an arrow at the end illustrates an example swing foot trajectory in response to the tripping event. (a) Early tripping over obstacles. (b) Premature (i.e., early) contact with the ground. (c) Late tripping over obstacles.

walking gaits used in this thesis are designed to lower the swing foot near the end of the gait to initiate leg swapping, the swing leg will be lowered making contact with the ground, and, at the ensuing step, a recovery controller focusing on rejecting the tripping disturbance of the previous step is applied. The parameter value in the recovery controller is chosen by optimization, and the detailed optimization process is explained next.

Design of the Recovery Controller

The design of the recovery controller involves selecting the following parameters, which are grouped into a vector denoted by,

$$\Theta^{\mathrm{RC}} := [\ \tilde{\alpha}_{\mathrm{T}}, k_{\mathrm{vc}}, k_{\mathrm{vd}}, q_{\mathrm{mLS_{vr}}}]^T, \tag{6.6}$$

where $\tilde{\alpha}_{\mathrm{T}}$ are coefficients of the Bézier polynomials for the torso virtual constraints except

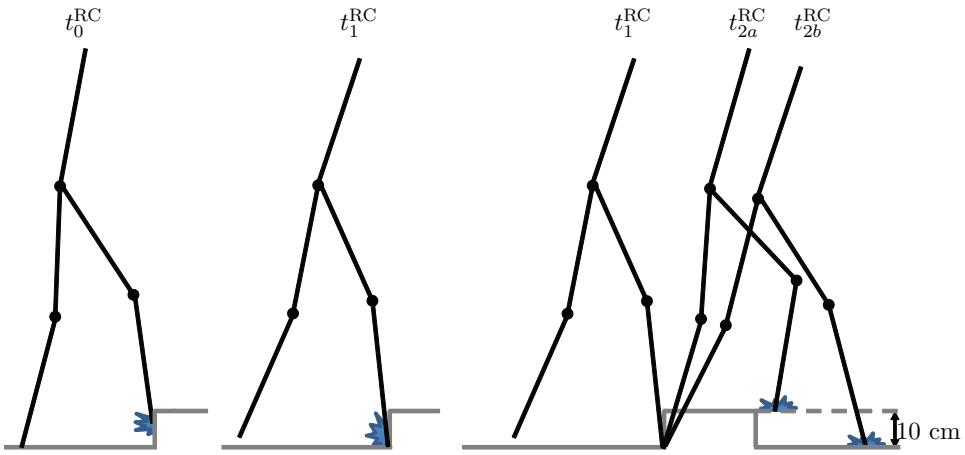


Figure 6.5: Design of recovery controller. At time t_0^{RC}, the robot has tripped over obstacles. Over $[t_0^{\mathrm{RC}}, t_1^{\mathrm{RC}})$, controller is kept unchanged, and swing leg lowered to make an impact with the ground at t_1^{RC}. Then, recovery controller (RC) will be applied at the ensuing step. The end time of the ensuing step depends on the shape and dimension of the obstacles. In scenario a, the robot has tripped over an wide obstacle, and the swing foot makes the contact on top of the obstacles at t_{2a}^{RC}. Whereas, in scenario b, the robot has tripped over a thin obstacle, so swing foot crosses over the obstacle and has the impact with the ground.

for the last coefficient (The last coefficient of the torso virtual constraint for the baseline controller is retained in order to provide a smooth transition to the baseline phase), and k_{vc}, k_{vd}, and $q_{\mathrm{mLS_{vr}}}$ are stiffness, damping, and rest position of the virtual compliance, respectively. Parameter values in (6.6) are obtained through iterative optimization. The optimization process is based on the simplified design model as in the design process of the step-down and step-up controllers.

Assume that the robot has tripped over an obstacle as shown in Figure. 6.5. Let t_0^{RC} be the time that the robot trips over the obstacle, let t_1^{RC} be the time the swing leg impacts the ground, and let t_2^{RC} be the end of next step. Over the interval $[t_0^{\mathrm{RC}}, t_1^{\mathrm{RC}})$, the baseline controller is applied, whereas over interval $[t_1^{\mathrm{RC}}, t_2^{\mathrm{RC}}]$, the recovery controller is applied. Because the swing foot slips down along the wall of the obstacle over the interval $[t_0^{\mathrm{rc}}, t_1^{\mathrm{RC}})$ while the stance foot is on the ground, the trajectory cannot be simulated using the simplified design model, which cannot account for double-support dynamics. Therefore, the detailed model is used for the simulation of this slip behavior during double support, and the value of the robot's state at t_1^{RC} after the impact with the ground, $x_{\mathrm{s}}\left(t_1^{\mathrm{RC}}\right)$, is acquired

from the detailed model simulation data.

Taking $x_{\rm s}\left(t_1^{\rm RC}\right)$ as an initial condition, the ensuing step can be obtained by simulation of the simplified design model, but the time at the end of the step varies depending on the sizes of the obstacles as shown in Figure 6.5. In this optimization problem, the following two scenarios are considered. In the scenario a, the robot has tripped over a 10 cm high obstacle and the swing foot is supported on the top of the obstacle at the end of the step. We can define $t_{2a}^{\rm RC}$ to be the time when the swing foot touches the top of the obstacle. In the scenario b, we assume that the robot has tripped over an obstacle with a very small width, and the swing foot touches the ground at the end of the step. We can define $t_{2b}^{\rm RC}$ to be the time when the swing foot touches the ground as shown in Figure 6.5.

The same cost function and constraints as used in Section 5.2 to obtain the step-down controller are also used here, with the following additional constraints:

(a) magnitude of torso angular velocity at the end of the step,

$$\left|\dot{q}_{\rm T}(t_{2b}^{\rm RC})\right| < \gamma_{\dot{q}_{{\rm T},rc}}, \tag{6.7}$$

in order to attenuate torso movement in preparation for impact,

(b) torso angle at the end of the step bounded below by some constant $\gamma_{q_{{\rm T},rca}}$,

$$q_{\rm T}(t_{2b}^{\rm RC}) > \gamma_{q_{{\rm T},rc}}, \tag{6.8}$$

to keep the torso from pitching forward too much at the end of the step,

(c) torso angular velocity at $t_{2a}^{\rm RC}$,

$$\dot{q}_{\rm T}(t_{2a}^{\rm RC}) \leq 0, \tag{6.9}$$

so that the torso has enough forward pitching velocity just before impact in order not to fall backward after impact with a 10 cm high obstacle.

Numerical optimization with the setting of $\delta = 0.1$, $\gamma_{\dot{q}_{\rm T}} = 240$ deg/sec, $C = 300$ N, $\mu_s = 0.6$, $\gamma_{\dot{q}_{{\rm LA}_{\rm sw},{\rm RC}}} = -340$ deg/sec, $\gamma_{\dot{q}_{{\rm T},{\rm RC}}} = 115$deg/sec, $\gamma_{q_{\rm T}} = -10$ deg, yields the

parameter vector,

$$\Theta^{\mathrm{RC}} = [-0.0910, -0.1702, 0.0655, -0.2912, -0.1901, 1.2421, 0.3, 2.7]^T . \tag{6.10}$$

After the virtual constraints for the torso and the parameters in the virtual compliance have been chosen through optimization, the virtual constraints for the swing leg are tuned by hand. The followings are considered for tuning the virtual constraints:

(a) swing leg angle is monotonically increasing in the middle of the gait,

$$\min \left\{ \dot{q}_{\mathrm{LA_{sw}}}(t) \,\middle|\, s_1^{\mathrm{RC}} \leq s(q_{\mathrm{s}}) < s_2^{\mathrm{RC}} \right\} > 0; \tag{6.11}$$

(b) swing leg angle velocity is bounded below by some negative value $\gamma_{\dot{q}_{\mathrm{LA_{sw}}}} < 0$ in the beginning of the step,

$$\min \left\{ \dot{q}_{\mathrm{LA_{sw}}} \,\middle|\, s(q_{\mathrm{s}}) < s_1^{\mathrm{RC}} \right\} > \gamma_{\dot{q}_{\mathrm{LA_{sw}}}}, \tag{6.12}$$

in order to keep the swing leg from going backward too far;

(c) swing toe vertical position over interval $s(q_{\mathrm{s}}) \in [s_1^{\mathrm{RC}}, s_2^{\mathrm{RC}}]$ should be higher than 10 cm in order to keep the swing leg from scuffing with the ground before $s(q_{\mathrm{s}})$ reaches s_2^{RC},

$$\min \left\{ p_2^h(q_{\mathrm{s}}) \,\middle|\, s_1^{\mathrm{RC}} < s(q_{\mathrm{s}}) < s_2^{\mathrm{RC}} \right\} > 0.1; \tag{6.13}$$

(d) swing leg shape at t_{2b}^{RC},

$$20\,\mathrm{deg} < q_{\mathrm{LS_{sw}}}(t_{2b}^{\mathrm{RC}}) < 26\,\mathrm{deg}, \tag{6.14}$$

so that the knee is neither too bent nor too straight just before impact with a 10 cm high obstacle.

Figure 6.6 displays results from the simplified design model simulation from t_1^{RC} to t_{2b}^{RC} of the recovery controller with optimized Θ^{RC} and the baseline controller. By using the

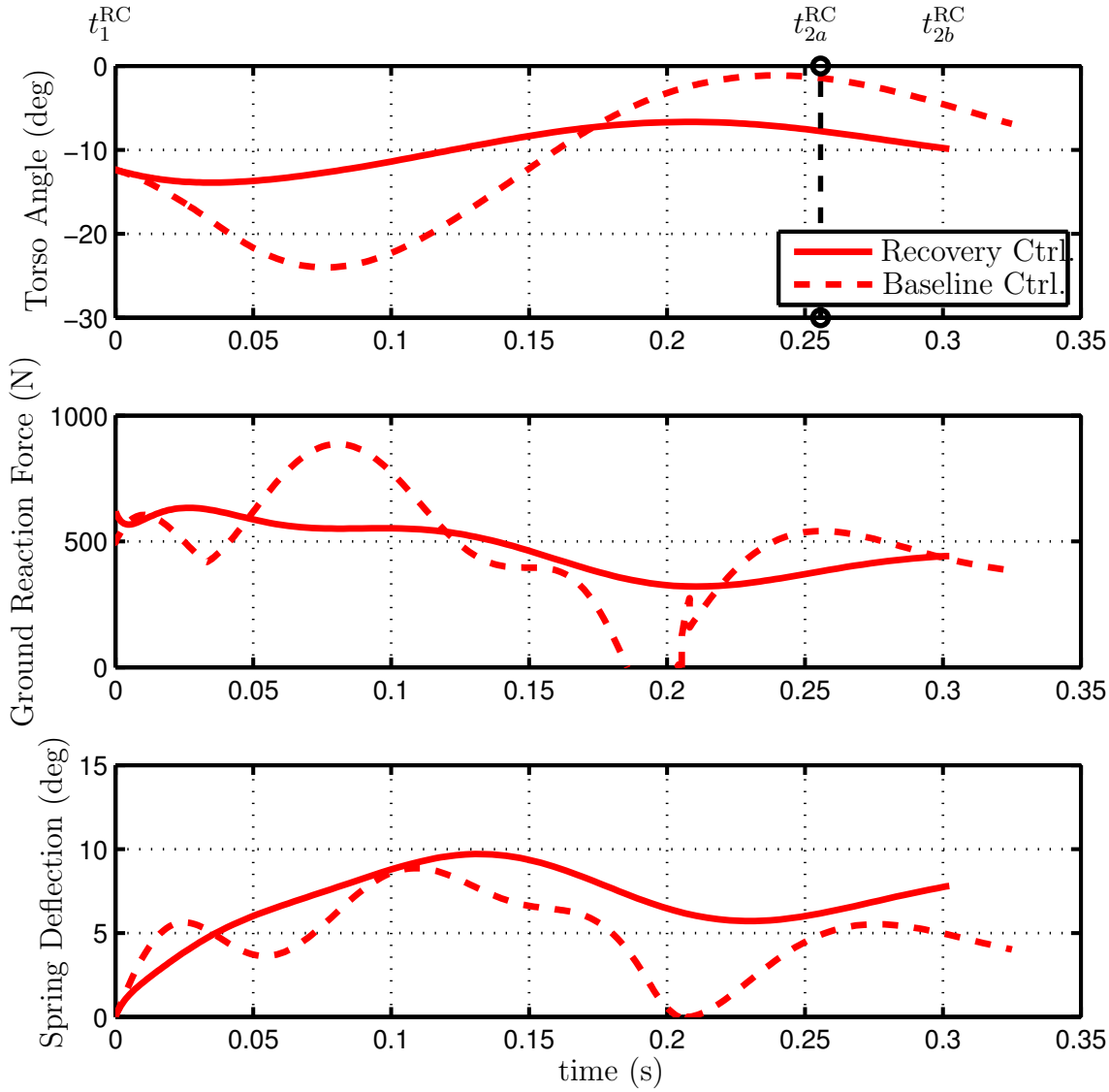


Figure 6.6: Simulation data with optimal parameter vector Θ^{RC} from the simplified design model for a trip recovery. The recovery controller is shown with a solid line, while a dashed line represents the baseline controller.

recovery controller, torso oscillation is not only reduced by approximately factor of 3, but also both ground reaction force and spring deflection remain positive during the recovery phase. Time t_{2a}^{RC} is indicated by the vertical black dotted line. It is observed that at time t_{2a}^{RC}, torso is pitching forward, satisfying constraint (6.9).

6.2.2 Reflexive Strategy for Early Tripping

Early tripping can be defined mathematically with respect to the robot's state and contact switch signals. Referring to Figure 6.4, impact with the ground ($sw_{\mathrm{toe}} = 1$) or an obstacle ($sw_{\mathrm{sh}} = 1$) is defined to be early tripping if it occurs in the middle of the gait ($s(q_{\mathrm{s}}) < s_{\mathrm{early}}$) or with swing leg not advanced adequately (absolute swing leg angle being less than some value, that is, $q_{\mathrm{LA_{sw}}}^{abs} := q_{\mathrm{LA_{sw}}} + q_{\mathrm{T}} < q_{\mathrm{early}}$). Mathematically, it is equivalent that the robot's state enters the following set:

$$\begin{aligned}\mathcal{S}_{\mathrm{RE}} := &\left\{ x_{\mathrm{s}} \in T\mathcal{Q}_{\mathrm{s}} \,\middle|\, s(q_{\mathrm{s}}) < s_{\mathrm{early}} \text{ or } q_{\mathrm{LA_{sw}}}^{abs}(q_{\mathrm{s}}) < q_{\mathrm{early}} \right\} \cap \\ &\left\{ (sw_{\mathrm{sh}}, sw_{\mathrm{toe}}) \in \{0,1\}^2 \,|\, sw_{\mathrm{sh}} = 1 \text{ or } sw_{\mathrm{toe}} = 1 \right\}. \end{aligned} \tag{6.15}$$

In response to early tripping, the rapid-elevation (RE) strategy of the swing leg is activated. Rapid-elevation strategy of the swing leg is accomplished by rapidly bending the swing knee by repositioning the leg shape motor of the swing leg. This is done through the appropriate choice of $h_{\mathrm{mLS_{sw}}}^{d,v}(\theta(q_{\mathrm{s}}), \alpha_v)$ in (5.3), so that the swing leg will clear a 10 cm obstacle. The details are given in Appendix F. The purpose of the elevation strategy of the swing leg is to place the swing foot on top of the obstacle, and allow the robot to continue the walking gait. Once the swing foot is on top of the obstacle, the robot can continue the walking by applying step-up controller at the ensuing step.

If the swing foot gets stuck on the vertical step and fails to be placed on top of the obstacle at the end of the step in spite of application of the rapid-elevation of the swing leg, the recovery controller which is designed in Section 6.2.1 will be applied at the ensuing step in order to have a safe continuation of the walking gait. Failure of rapid-elevation strategy of the swing leg can be detected by checking the calculated height of the swing foot at the end of the gait, denoted by $p_{\mathrm{toe_{sw}}}^{v}(q_{\mathrm{s}})$, and how much the swing foot advanced

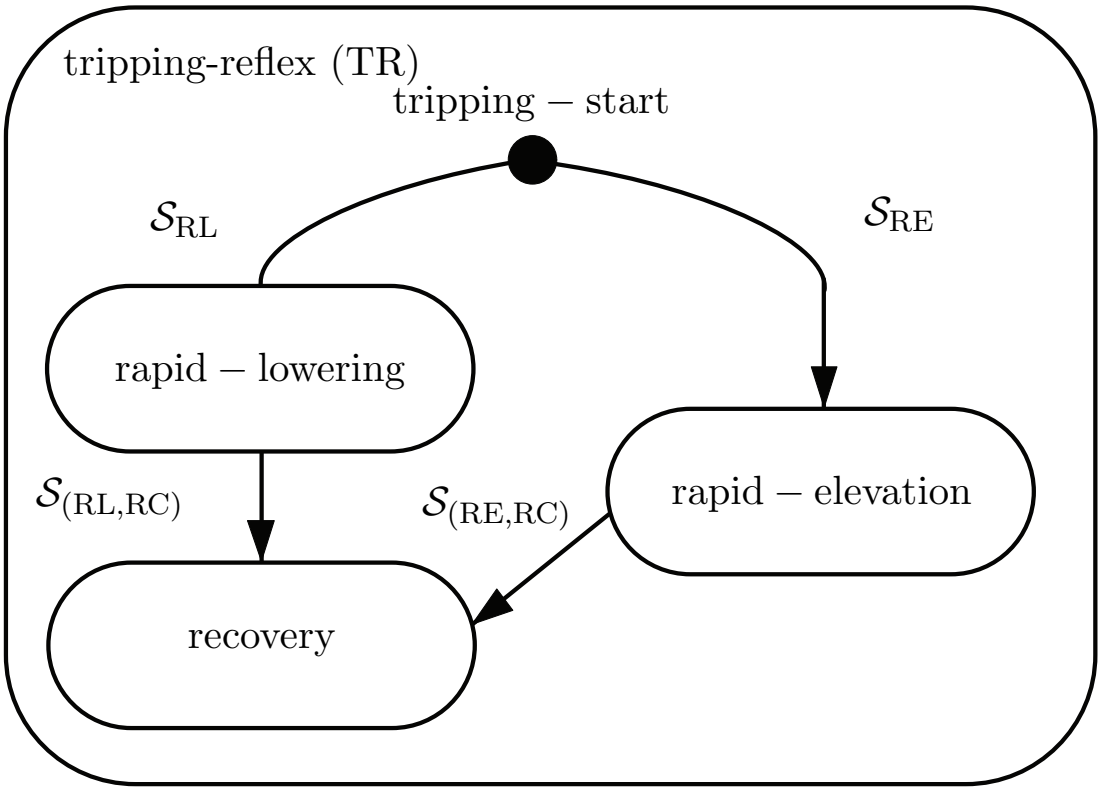


Figure 6.7: Finite-state machine of tripping-reflex controller

horizontally over $[t_0^{\mathrm{rc}}, t_1^{\mathrm{RC}})$, denoted by $\delta p^h_{\mathrm{toe_{sw}}}$; if the swing foot gets stuck on the vertical step, the swing foot will not be able to advanced forward ($\delta p^h_{\mathrm{toe_{sw}}} \approx 0$) and will make an impact with the ground instead of on top of the obstacle ($\left|p^v_{\mathrm{toe_{sw}}}(q_{\mathrm{s}})\right| \leq \Delta H$). Combining all of these conditions yields the following set given by,

$$\mathcal{S}_{(\mathrm{RE,RC})} := \left\{ x_{\mathrm{s}} \in \mathcal{S}^H \,\middle|\, s(q_{\mathrm{s}}) \geq s_{\mathrm{early}},\ q^{abs}_{\mathrm{LA_{sw}}}(q_{\mathrm{s}}) \geq q_{\mathrm{early}}, \left|p^v_{\mathrm{toe_{sw}}}(q_{\mathrm{s}})\right| \leq \Delta H, \left|\delta p^h_{\mathrm{toe_{sw}}}\right| < c_{\mathrm{fail}} \right\}, \tag{6.16}$$

where $c_{\mathrm{fail}} \approx 0$ is a scalar constant.

The tripping-reflex controller is summarized in Figure 6.7 including rapid-lowering and rapid-elevation of the swing leg in response to late and early tripping, and activation of the recovery controller after the failure of rapid-elevation of the swing leg strategy.

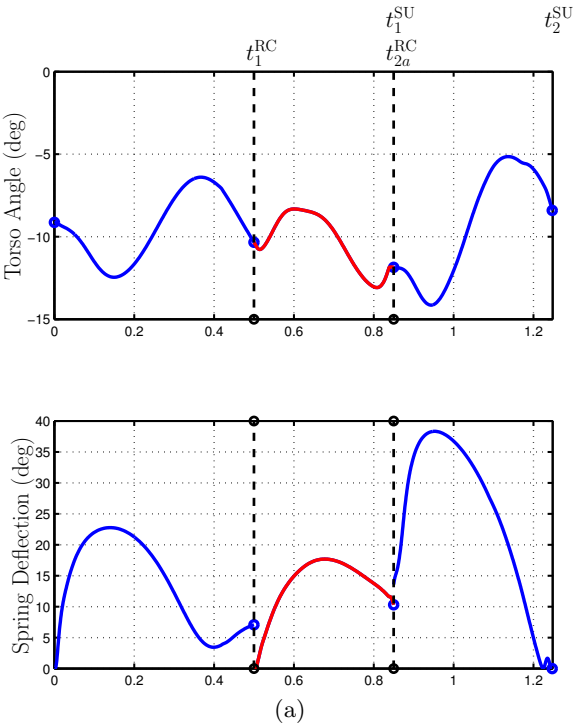
6.3 Controller Verification of Detailed Model

The proposed controllers, the step-up controller and the tripping-reflex strategies for late and early tripping (rapid-lowering of the swing leg and rapid-elevation of the swing leg), are verified on the detailed model. With the modifications explained in Section 5.3 to the proposed controller, the simulation results in MABEL successfully accommodating a 10.0 cm vertical step. Here, three different scenarios are considered to verify the performance of both the step-up controller and the tripping-reflex strategies, simultaneously. In the first two scenarios, the robot trips over the vertical step in late and early swing phase, while, in the last scenario, the rapid elevation of the swing leg in response to early tripping has failed, and recovery control is applied at the ensuing step. Figures 6.8, 6.9, and 6.10 show simulation data of the three scenarios. In each scenario, the robot successfully stepped onto the 10.0 cm platform. Neither large torso oscillation nor excessive unloading of the spring is observed. (see Figures 6.8a, 6.9a, and 6.10a)

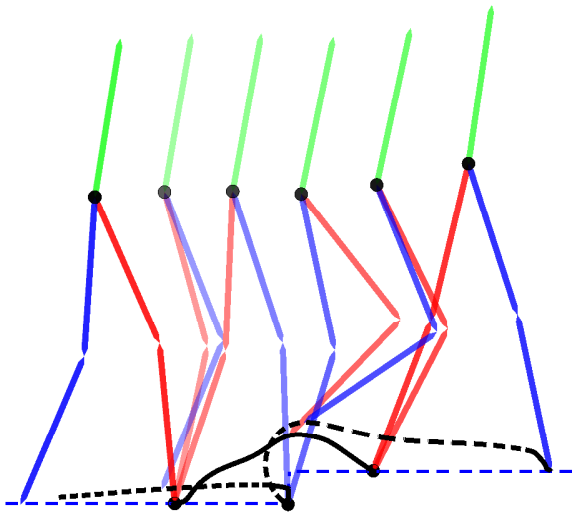
6.4 Experiment

The step-up controller, and the tripping-reflex strategy are now evaluated on the robot. A platform with height of 10.5 cm and length 1.9 m is constructed. In this experiment, MABEL stepped upon the platform with a height of 10.5 cm (more than 10% of its leg length), walked three steps on level ground on top of the platform, and stepped off the platform; see Figure 7.7. Two laps were completed before the robot was stopped by a researcher. On the first lap, MABEL's swing foot struck the platform at a late point of the step, so the rapid-lowering controller was activated in response to the late perturbation (see Figure 6.12(a)). On the other hand, on the second lap, the rapid-elevation controller is triggered in response to the early perturbation as can be seen in Figure 6.12(b). By using either controller, MABEL could successfully step upon and walk for 3 steps across a 10.5 cm platform. We verified that MABEL could ascend a platform with a height of up to 12.5 cm, which is 12.5% of leg length (screen capture of the video of the experiment is provided in Appendix G.).

We observe that the rapid-elevation controller is unnecessarily activated from 1.8 sec to



(a)

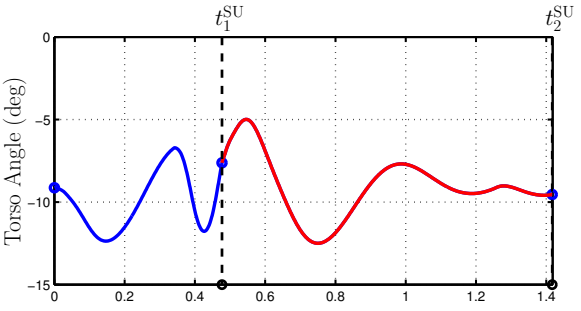


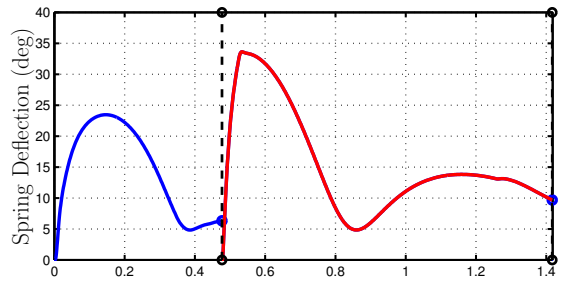
(b)

Figure 6.8: Simulation data of the first scenario (Recovery from Late Tripping). (a) Torso and spring deflection (b) Stick figure illustration. Black circle at the toe indicates stance toe while black lines indicate swing toe trajectories.

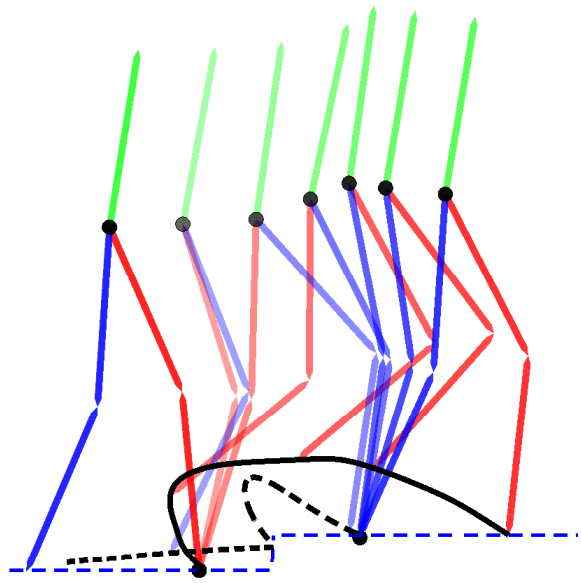
1.4 sec on the first lap. This unnecessary activation of the rapid-elevation controller were caused by a malfunction of the contact switches[1].

[1]This wrong activation of switches is caused by the vibration of switch covers that have non-negligible mass and inertia. During the gentle maneuvers, of MABEL, vibration of switch covers was not a problem because it had been efficiently damped out by the spring installed on the switch covers. However, very





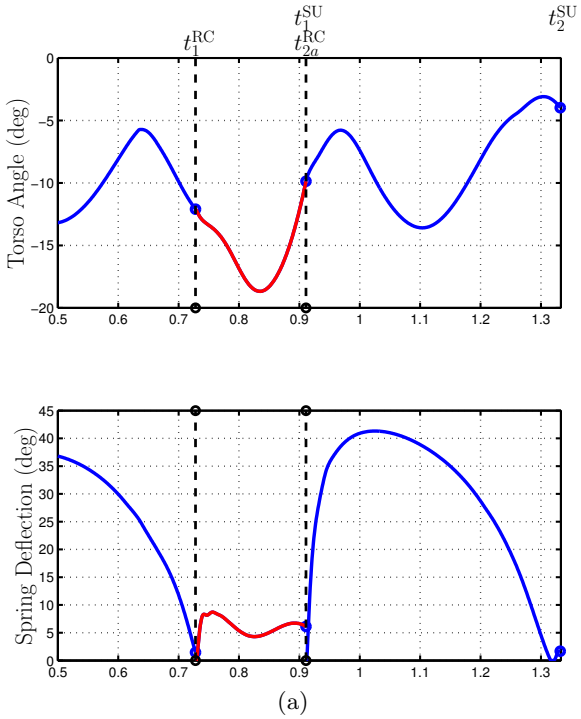
(a)



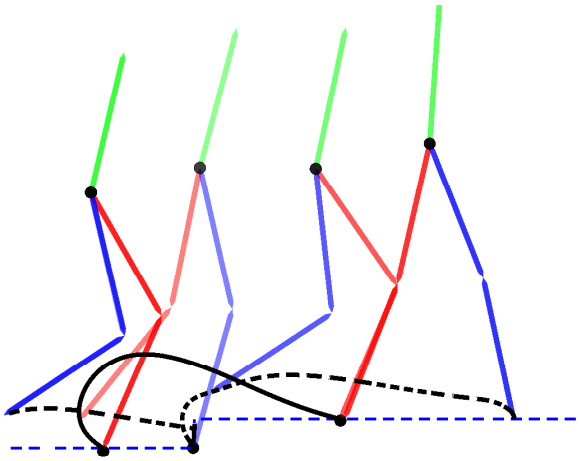
(b)

Figure 6.9: Simulation data of the second scenario (Recovery from Early Tripping). (a) Torso and spring deflection (b) Stick figure illustration. Black circle at the toe indicates stance toe while black lines indicate swing toe trajectories.

dynamic maneuvers such as walking over obstacles with considerable height, can cause an unexpected closing of contact switches due to the vibration of the switch. This small issue on the contact switches can be easily solved by a redesign of the contact switch mechanism, such as adding more damping on the springs.



(a)



(b)

Figure 6.10: Simulation data of the third scenario (Failure of swing elevation). (a) Torso and spring deflection (b) Stick figure illustration. Black circle at the toe indicates stance toe while black lines indicate swing toe trajectories.

6.5 Conclusion

A step-up controller and tripping-reflex strategy have been designed to deal with blind step-ups of considerable height. Experimentation with this controller showed MABEL stepping onto a 12.5 cm platform. Our next work will focus on the proper combination of these

two step-up and step-down controllers to provide considerable robustness to uneven terrain.

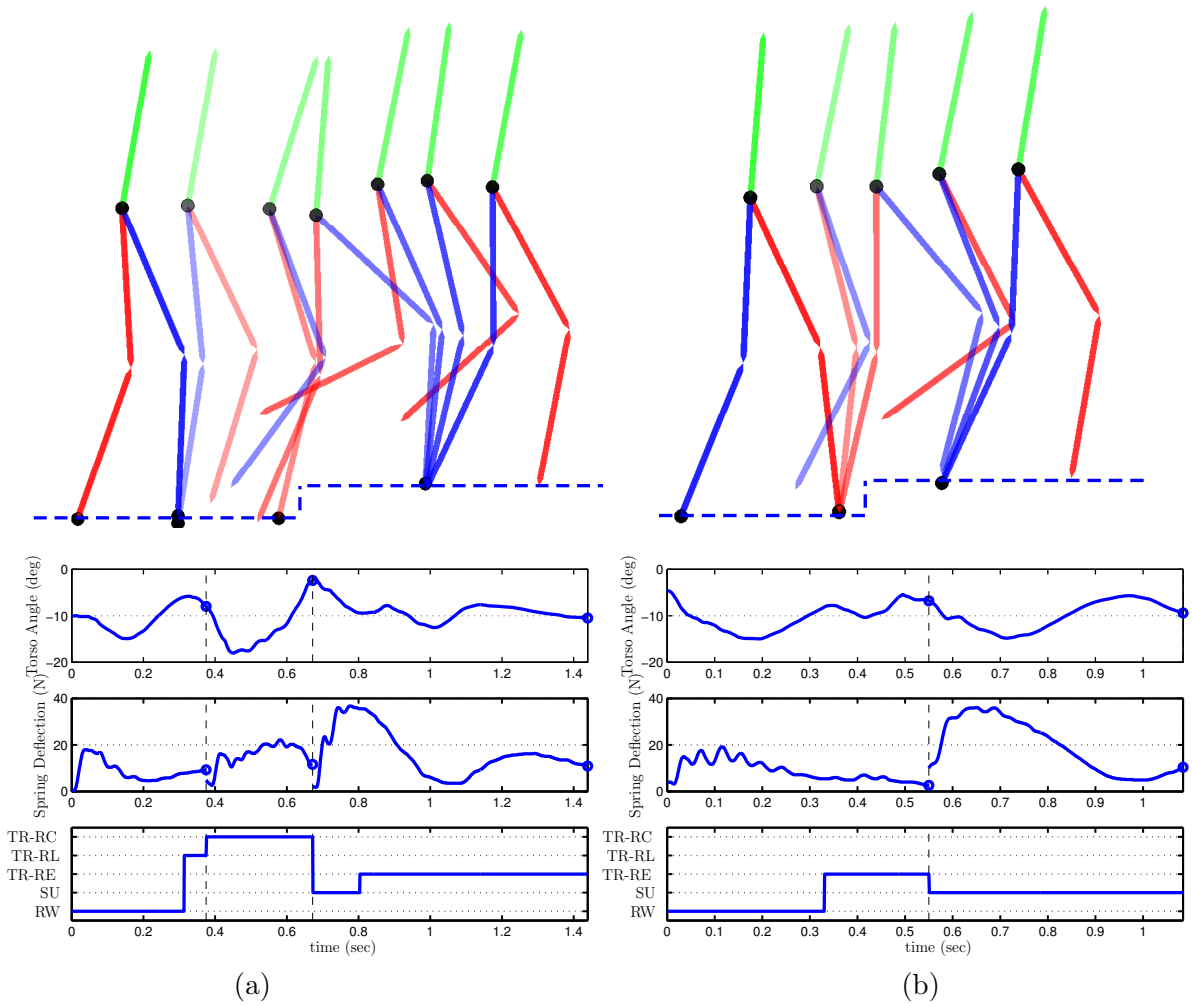
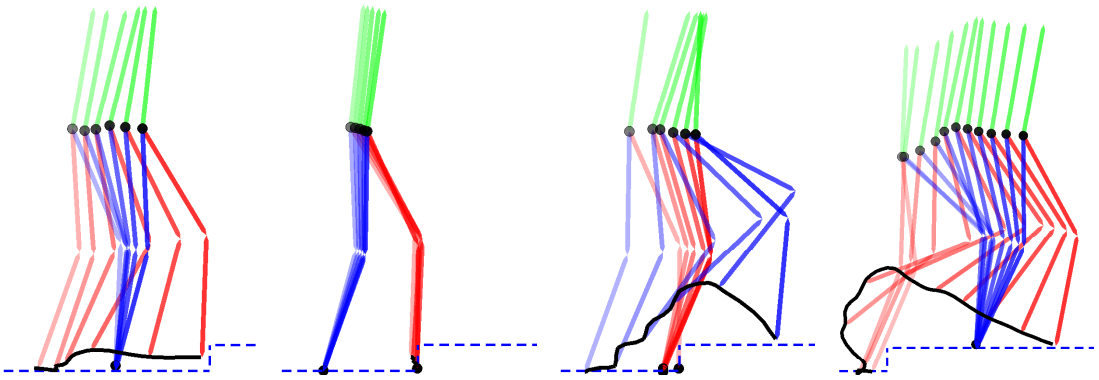
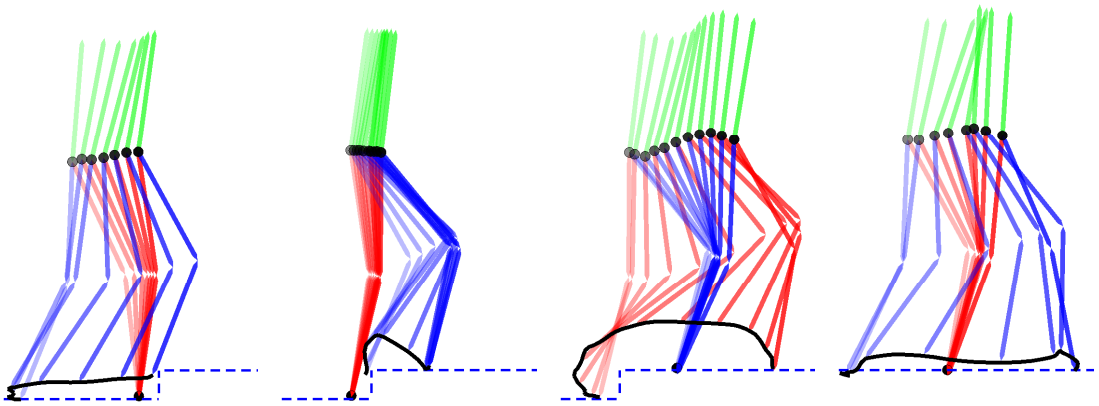


Figure 6.11: Experimental data of step-up of 10.5 cm platform. In stick figure illustration, black circle at the toe indicates stance toe. Torso angle (top), s (middle), and types of controllers (bottom) applied are shown next, respectively. In these figures, blue hollow circle and black dashed line indicate the end of the step. RW, SU, TR-RE, TR-RL, TR-RC refer walking on flat ground, step-up, rapid-elevation of the swing leg, rapid-lowering of the swing, and recovery, respectively. (a) First Lap (b) Second Lap



(a)



(b)

Figure 6.12: Stick figure illustration of experimental data of step-up and step-down for a 10.5 cm platform. Black circle at the toe indicates stance toe. (a) On the first lap, the rapid-lowering controller is activated in response to late tripping while (b) the rapid-elevation controller is triggered in response to early tripping on the second lap.

CHAPTER VII

A Finite-state Machine for Traversing Unexpected Large and Rapid Ground Height Variations

This chapter presents a finite-state machine that manages transitions among controllers for flat-ground walking, stepping-down, stepping-up, and a trip-reflex. If the robot completes a step, the depth of a step-down or height of a step-up can be immediately estimated at impact from the lengths of the legs and the angles of the robot's joints. The change in height can be used to invoke a proper control response. On the other hand, if the swing leg impacts an obstacle during a step, or has a premature impact with the ground, a trip reflex is triggered on the basis of specially designed contact switches on the robot's shins, contact switches on the end of each leg, and the current configuration of the robot. The design of each control mode was done in Chapters V and VI. Here, the transition conditions among them are presented. The control policy allows MABEL to traverse various ground profiles, including ascent of a stair with a height of 12.5 cm (12.5% of the leg length), step-down from a platform with a height of 18.5 cm (18.5% of the leg length)[1], and walking over constructed platforms which consist of steps with heights of 10.5 cm and 8.0 cm (see Figure 7.1), without falling. The robot is provided information on neither where a disturbance occurs, nor how extensive it is.

[1] In Chapter V, the implementation of a simple switching control policy with only step-down and baseline controllers enabled MABEL to step off an 8 inch (20.32 cm) platform. The 8 inch (20.32 cm) step-down test was not tried with the finite-state machine designed in this chapter. Instead, a step-down from an 18.5 cm high platform, which is only 2 cm less than the 20.32 cm platform, was tried and successfully accommodated by the finite-state machine.

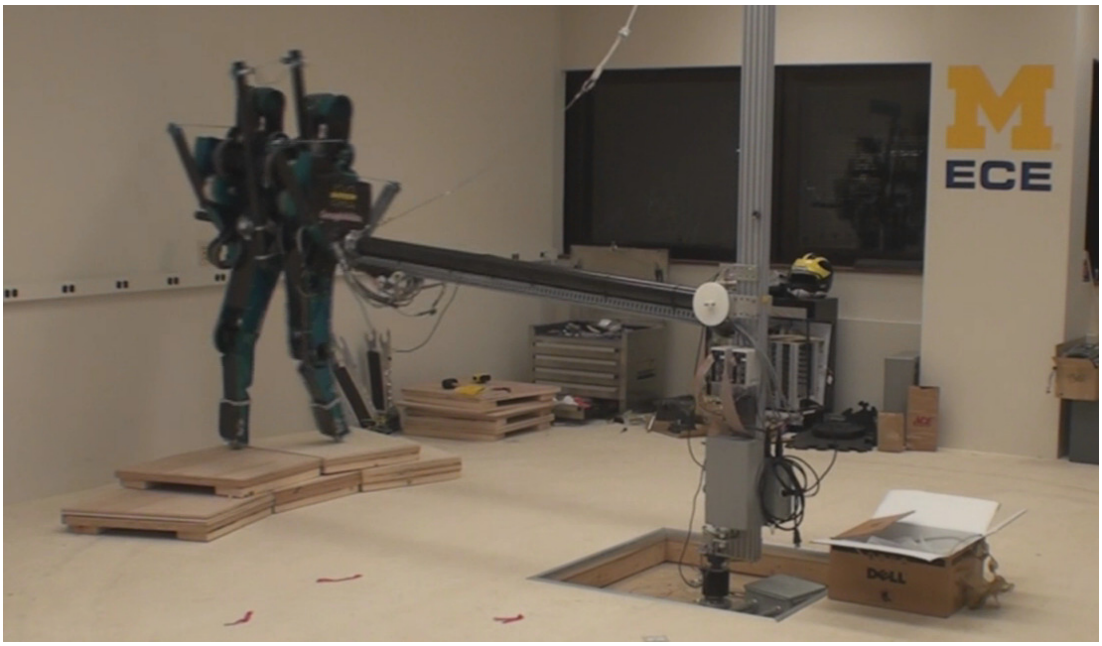


Figure 7.1: MABEL is traversing the platform with 10.5 cm and 8 cm steps.

7.1 The Regular-walking Controller

In Chapters V and VI, stiffness values of the virtual compliant element for step-down and step-up disturbances are chosen to be 1.2198 and 2.1088 respectively by optimization. These values for the stiffness of virtual compliant element, higher stiffness for step-down disturbances and lower stiffness for step-up disturbances, are well aligned with the values of effective stiffness used by humans when accommodating significant step-down and step-up disturbances. In particular, it is found that humans adjust the stiffness of the stance leg in response to ground variations in height during running [39, 68]. The adjustment is characterized by an increase in leg stiffness in preparation for a step-down disturbance and a reduction in response to a step-up disturbance compared to the leg stiffness used for running over flat ground. In MABEL, this idea of a reflex strategy in response to ground-height variations was implemented in the context of switching-control design through active force control. Changes in the effective leg stiffness were easily made by varying the values of stiffness k_{vc}, damping k_{vd}, and rest position $q_{\mathrm{mLS}_{vr}}$ in the virtual compliant element of the active force control.

Motivated by this similarity in strategies between MABEL and humans, we now extend

the use of active force control to walking over flat ground. This extension also agrees with the observation in [95] that the human's stance knee essentially shows the dynamic behavior of a spring-damper system when walking on the flat ground.

Therefore, the baseline controller introduced in Section 5.1 is modified to include active force control by replacing the motor leg shape control with a virtual compliance as in Section 5.2.2. We use this modified baseline controller for the walking on flat ground hereafter. The parameter values for the virtual compliance are iteratively found thorough simulations, and given by,

$$\alpha_{vc}^{\mathrm{RW}} := [1.8, \quad 0.1297, \quad 2.7]^T . \tag{7.1}$$

7.2 Categorization of Walking

The design process of the finite-state machine begins with the definition of four different types of stance phase with respect to perturbations in ground height, or disturbances from tripping over obstacles: regular-walking phase (RW), step-down phase (SD), step-up phase (SU), and tripping phase (TR) (see Figure 7.2). Hereafter, we denote phases by elements of the following index set,

$$W := \{\mathrm{RW}, \mathrm{SD}, \mathrm{SU}, \mathrm{TR}\} . \tag{7.2}$$

The decisions to transition among phases will be made on the basis of a change in walking surface height, which can be immediately computed at impact from the length of the robot's legs and the angles of its joints, along with the values of the contact switches. Therefore, switching surfaces which implement the transitions among phases must be designed as shown in Figure 7.2.

Firstly, the transition to RW takes place when the impact with the ground ($x_{\mathrm{s}} \in \mathcal{S}^H$) occurs close to the end of the gait ($s(q_{\mathrm{s}}) \geq s_{\mathrm{early}}$) as well as when the magnitude of the height of the swing toe at the moment of the impact is less than some scalar threshold value ΔH, which is set to 0.05m in this chapter ($\left|p^v_{\mathrm{toe}_{\mathrm{sw}}}(q_{\mathrm{s}})\right| \leq \Delta H$). Mathematically, it is

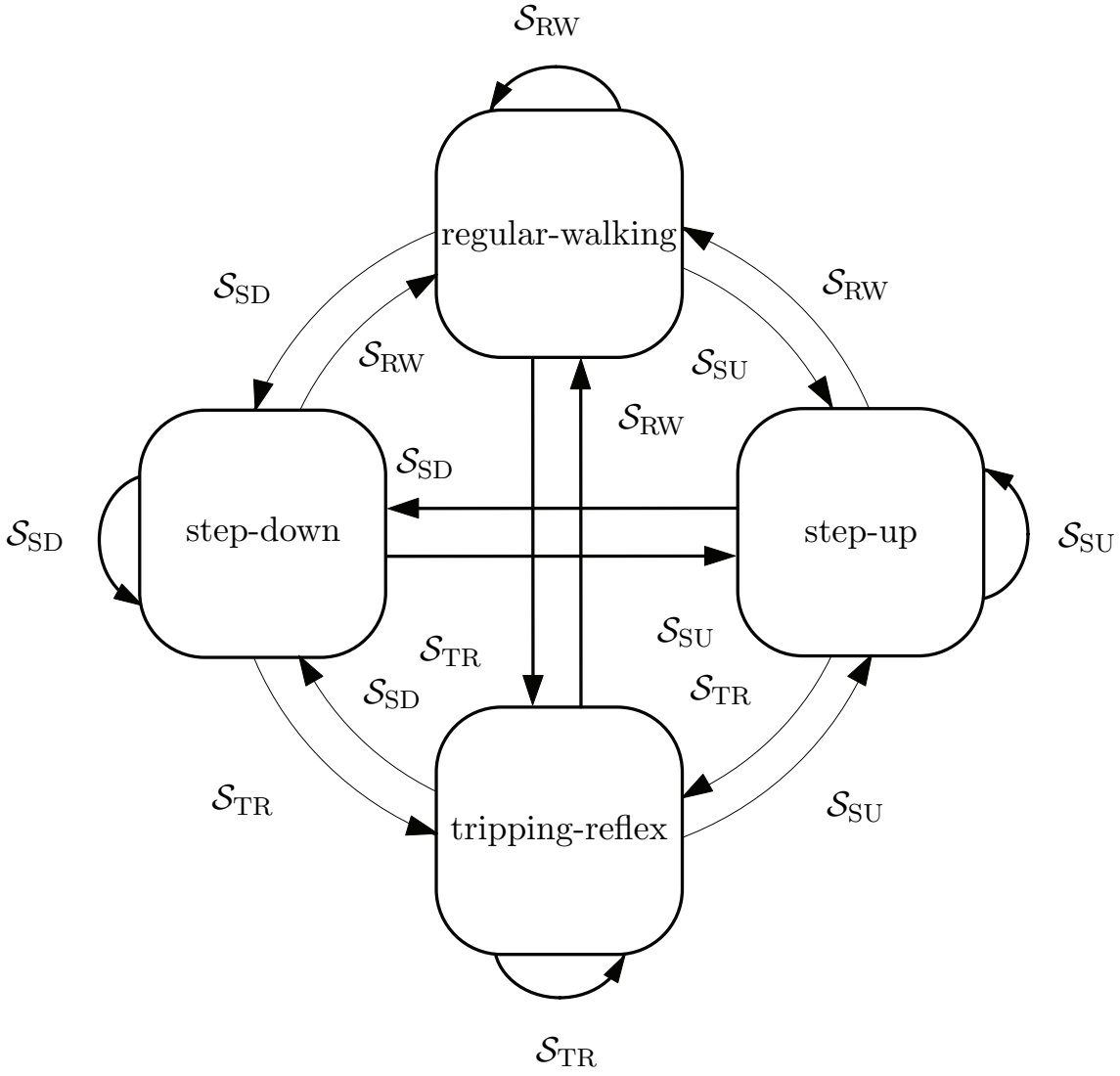


Figure 7.2: Phases and associated switching surfaces of switching control design.

equivalent that the robot's states enter the following switching surface,

$$\mathcal{S}_{\mathrm{RW}} := \left\{ x_{\mathrm{s}} \in \mathcal{S}^H \;\middle|\; \left| p^v_{\mathrm{toe_{sw}}}(q_{\mathrm{s}}) \right| \leq \Delta H, s(q_{\mathrm{s}}) \geq s_{\mathrm{early}} \right\}. \tag{7.3}$$

The transition to SD or SU occurs when the impact with the ground occurs close to the end of the gait ($s(q_{\mathrm{s}}) \geq s_{\mathrm{early}}$), along with the height of the swing toe at the moment of the impact being less than $-\Delta H$ or larger than ΔH respectively. Mathematically, it is equivalent that robot's state enters the switching surface $\mathcal{S}_{\mathrm{SD}}$ for detecting step-down

disturbances

$$\mathcal{S}_{\mathrm{SD}} := \left\{ x_{\mathrm{s}} \in \mathcal{S}^H \,\middle|\, p^v_{\mathrm{toe_{sw}}}(q_{\mathrm{s}}) < -\Delta H, \Delta H \in \mathbb{R}, s(q_{\mathrm{s}}) \geq s_{\mathrm{early}} \right\}, \tag{7.4}$$

or the following switching surface for detecting step-up disturbances

$$\mathcal{S}_{\mathrm{SU}} := \left\{ x_{\mathrm{s}} \in \mathcal{S}^H \,\middle|\, p^v_{\mathrm{toe_{sw}}}(q_{\mathrm{s}}) > \Delta H, \Delta H \in \mathbb{R}, s(q_{\mathrm{s}}) \geq s_{\mathrm{early}} \right\}. \tag{7.5}$$

Lastly, the transition to TR arises when the swing leg trips over obstacles or touches the ground unexpectedly. Tripping can be detected by checking both contact switch signals installed on the front and bottom of the shin of the swing leg along with the robot's states. The mathematical definition of the switching surface for the tripping phase is given as $\mathcal{S}_{\mathrm{TR}} := \mathcal{S}_{\mathrm{RL}} \cup \mathcal{S}_{\mathrm{RE}}$, where $\mathcal{S}_{\mathrm{RL}}$ is switching surface for detecting late tripping as defined in (6.5) and $\mathcal{S}_{\mathrm{RE}}$ is switching surface for detecting early tripping as defined in (6.15).

For each phase $w \in W$, the following four controllers are applied: regular-walking controller, step-down controller designed in Chapter V, and step-up controller and tripping-reflex controllers designed in Chapter VI. Furthermore, a finite-state machine to manage the transitions among phases is introduced as shown in Figure 7.2. Switching between the controllers takes place at the moment when transition between phases occurs.

7.3 Controller Evaluation on the Detailed Model

Before experimental deployment, the finite-state machine will be simulated on the detailed model. Certain straightforward modifications to the regular-walking, step-down, step-up, and tripping-reflex controllers are required due to discrepancies between the simplified and detailed models. Initial simulations will reveal one additional modification that needs to be performed. After these changes to the controllers, the performance of the finite-state machine will be evaluated when the controllers are sequentially composed in response to the disturbances in ground height.

7.3.1 Minor Modification of Controllers for Detailed Model Implementation

As part of implementing the proposed controllers on the detailed model, the following modifications are made to compensate for the gap between the simplified design model and the detailed model.

7.3.1.1 Modification for Cable Stretch

The most critical reason for model discrepancy is cable stretch. To account for the stretching of the cables, the coefficients of the virtual compliance k_{vc} in regular-walking, step-down, step-up, and tripping-reflex controller are modified as in [98, 99] so that the series connection of the compliance due to the cable stretch and the virtual compliance has the effective compliance specified by the optimization process. The details are given in Appendix E.

7.3.1.2 Modification for Asymmetry

In the experimental setup, due to the boom, the robot's hip position is constrained to lie on the surface of a sphere, rather than a plane. The hip width (distance between the legs) being 10% of the length of the boom causes the robot to weigh 10% more when supported on the inner leg (almost 70 N) than when supported on the outer leg. This causes inner-outer asymmetry in the walking gait [100]. To account for this asymmetry, the virtual compliance is made an additional 10% stiffer on the inside leg.

7.3.1.3 Modification for Avoiding Foot Scuffing

As discussed in Sections 6.1 and 6.2, the step-up and tripping-reflex controllers use the virtual constraints for the swing leg with a modification allowing the swing leg height to be increased in order to keep the swing foot from scuffing the ground. Similar modifications are required for step-down and regular-walking controllers.

(a) Step-down Controller: When MABEL steps off platforms, the higher impact force causes an additional bend in the stance knee at the ensuing step of the step-down event. The higher the platform which MABEL steps off is, the greater the bend in the stance knee

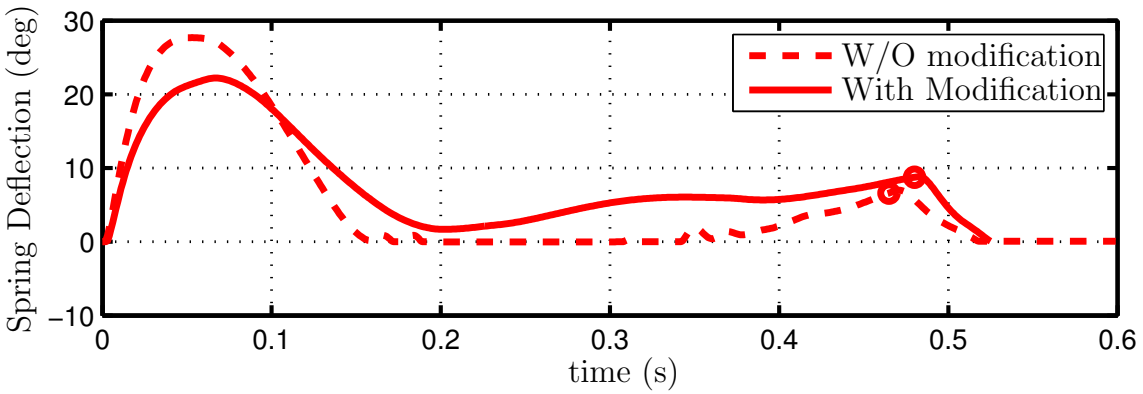


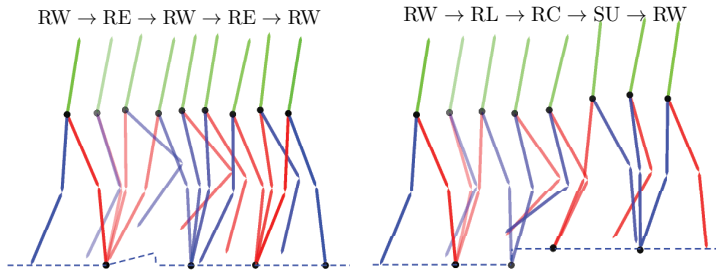
Figure 7.3: Simulation data of a step with the stance knee being excessively bent at the start. The dashed line shows simulation result without modification on the virtual compliance, and the solid line illustrates simulation result with modification. The small circles on the plot indicate the end of the step.

caused. To deal with this additional bend, the virtual constraint for the swing motor leg shape is increased according to the platform height calculated at impact.

(b) Regular-walking Controller: In the regular-walking phase, foot scuffing can also occur when a step starts with the stance knee being overly bent. Therefore, an event-based control is introduced so that the virtual constraint for the swing motor leg shape is increased according to how much more the stance knee is bent from some reference value. In particular, if the stance leg shape angle is larger than 10 deg at the start of the gait, the desired swing motor leg shape is then modified by increasing the middle two Bézier coefficients of the swing leg shape proportional to the difference between the stance leg shape angle and 10 deg.

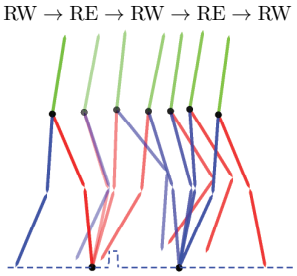
7.3.2 Additional Modification of the Virtual Compliance

The initial simulation of the controller showed that when the stance knee angle, which is two times the leg shape angle, is bent more than 60 deg at impact, the spring compresses further and subsequently decompresses very rapidly, causing the stance leg to loose contact with the ground. Figure 7.3 shows typical spring behavior when a step begins with the stance knee being excessively bent. The dashed line indicates the controller without modification. We can observe that the spring compresses further and then rapidly decompresses, and

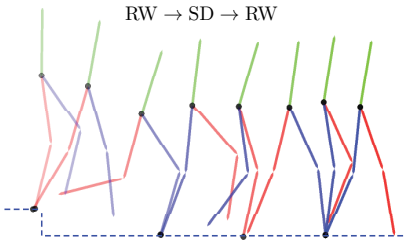


(a) Small Ramp

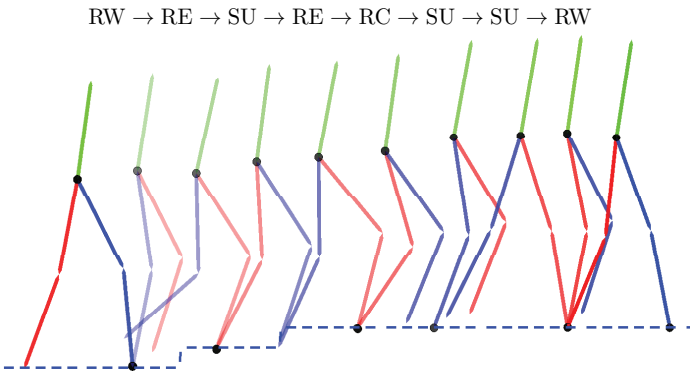
(b) Stair



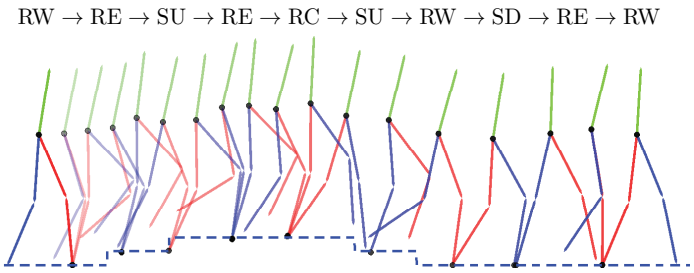
(c) Small Bump



(d) Step-down



(e) Two Consecutive Stairs



(f) Step-up and Step-down of Two Consecutive Stairs

Figure 7.4: Finite-state machine simulation on various ground profiles. Black circle at the toe indicates stance toe. The states of the finite-state machine are given above each stick figure illustration.

finally reaches zero over the interval from 150 msec to 350msec, causing the robot to lose contact with the ground.

This rapid change on the spring deflection is caused by the large amount of bend in the stance knee and the mechanical structure of the transmission mechanism. As in (5.7), the virtual compliance generates a control torque proportional to the deviation of the stance motor leg shape angle from the reference point. When a step begins with the stance knee being extremely bent, deviation from the reference point will be very large, creating large torque in the virtual compliance. Because virtual compliance and the physical spring are connected serially thorough the transmission mechanism, the spring will experience the same amount of torque as that which has been created by the virtual compliance. This large torque at the spring leads to rapid compression and decompression of the spring, causing loss of ground contact in the stance foot. In order to account for this problem, we soften the virtual compliance by the amount by which the stance knee is additionally bent from the nominal value. The solid line in Figure 7.3 illustrates spring deflection after the modification. With the modification, the spring compresses and decompresses gently, and does not reach zero.

7.3.3 Simulation of the finite-state machine

With these modifications to the proposed controller, the simulation results show MABEL successfully traversing various ground profiles representing large obstacles without *priori* information; in particular, vision feedback is not used. Figure 7.4 shows a stick figure illustration of the simulation results. The sequential composition of the controllers used in order to traverse the terrain is given above each stick figure. It is observed that the robot's configuration converges to the nominal configuration within a few steps after each disturbance event, and, in response to various disturbance events, appropriate switching between controllers occurs.

Figure 7.5 is a detailed illustration of Figure 7.4e. It is noted that different sequences are used at the first stair (RE $\rightarrow$ SU) and the second stair (RE $\rightarrow$ RC $\rightarrow$ SU). This is because, at the second step, the robot fails to place the swing toe on top of the stair (see Figure 7.5e) while, at the first stair, the robot was able to do so by using the rapid-elevation

controller (see Figure 7.5c).

7.4 Experiment

The proposed controller is now evaluated on the robot. In the experiments, walking over various types of platforms is tested to verify various executions of the finite-state machine.

7.4.1 Small Bump with Height of 7 cm

As a first test of the tripping-reflex controller, a small bump higher than the maximum vertical position of the swing foot was placed in the middle of MABEL's walking path. As shown in Figure 7.6, in response to tripping over this small bump, the tripping-reflex controller was activated, and modification of the swing foot trajectory was made successfully, thereby providing clearance of the bump.

7.4.2 Step-up and Step-down of 10.5 cm platform

In this experiment, MABEL walked up a platform with a height of 10.5 cm (more than 10% of its leg length), walked three steps on level ground on top of the platform, and stepped off from the platform; see Figure 7.7. Two laps were completed before the robot was stopped by a researcher. Figure 7.7 shows experimental results.

7.4.3 Consecutive Two Step-up and One Large Step-down of 18.5 cm platform

For the next experiment, we built a platform with two stair-steps, one with a height of 10.5 cm and the other with a height of 8 cm as shown in Figure 7.8. MABEL walked up two consecutive risers, took two steps of level walking on top of the platform, and stepped off from the platform with one step (18.5 cm step-down, which is 18.5% of the leg length), while skipping a stair-step placed between the top of the platform and the ground. The bottom graph of Figure 7.8b shows the horizontal hip velocity. In the graph, the horizontal hip velocity at the end of the two stair ascents is 0.9 m/sec (see the blue circle at 1.295 sec in the hip velocity graphs.) which is negligibly different from 0.93 m/sec of the average flat-walking hip velocity measured at the end of the step. After step-down from the 18.5 cm

high platform, hip velocity at the end of step was increased to 1.8 m/sec (see the blue circle at 3.148 sec), but reduced to 1.12 m/sec rapidly at the ensuing step. From these analyses of the hip velocity, it is observed that hip velocity is successfully regulated by the proposed controller in response to large step-up and step-down disturbances.

The rapid-elevation controller in the middle of the step-down phase is activated from 2.919 sec to 3.148 sec because the swing foot struck a stair between the top of the platform and the ground during a step-down event.

A similar experiment was conducted on the same platform, but MABEL took two steps during the step-down as shown in Figure 7.9. Therefore, the step-down controller was activated twice in a row in this case.

7.4.4 One Step-up, One Regular-walking Step, and One Step-down of 10.5 cm Platform

In this experiment, MABEL stepped upon a 10.5 cm platform, took one step on the top of the platform, and stepped off the platform as shown in Figure 7.10. Figure 7.11 shows a stick figure illustration of the step-down phase in Figure 7.11a and the ensuing step in Figure 7.11b, along with torso angle, spring deflection, and the types of controllers applied during the experiment in Figure 7.11c. As shown in Figure 7.11, tripping over the ground occurred at the end of the step. In response to the tripping event, the rapid-elevation controller was applied (see 0.411 sec of Figure 7.11c), but the swing foot could not clear the ground. Therefore, the recovery controller was triggered at the ensuing step instead of the regular-walking controller. This clearly shows use of the transition from rapid-elevation phase to recovery phase as shown in Figure 6.7.

7.4.5 Discussion on the Experiments

Although MABEL was able to accommodate various combinations of terrain as explained above, several tests conducted on certain combinations of disturbances showed limitations in the finite-state machine. For example, MABEL always failed to accommodate a platform with consecutive step-up and step-down disturbances although experiments on this type of ground variations were repeated several times. Similar results were obtained

by other experiments on a platform with consecutive step-down and step-up disturbances. Typical examples of platforms are shown in Figure 7.12. It can be conjectured that the reason for these failures is that the robot's state after the step-up or step-down disturbance is not in the "basin of attraction" of the ensuing step-down or step-up controller. Quantifying these basins of attraction and redesigning the step-up or step-down controllers are extremely challenging tasks.

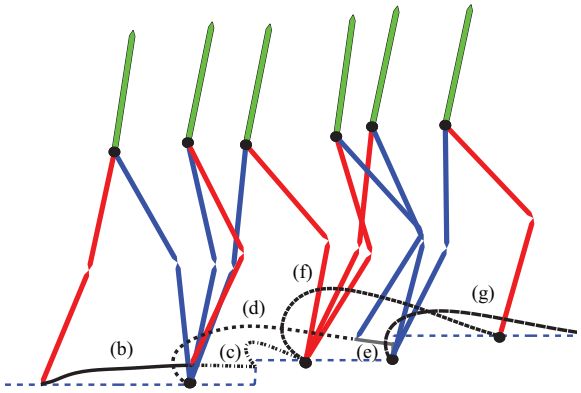
7.5 Conclusion

A finite-state machine has been designed to handle various kinds of disturbances including blind step-downs and step-ups of considerable height, and tripping over obstacles. The proposed finite-state machine consists of a regular-walking controller for walking on flat ground and three different controllers, which are a step-down, step-up, and tripping-reflex controller, whose objectives are the safe continuation of walking in response to a large disturbance in ground height and tripping over obstacles.

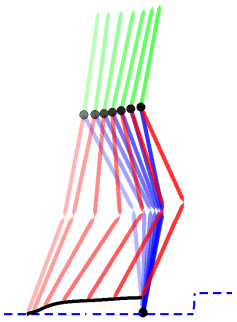
Each of the controllers employs active force control at the stance knee, with the feature of being able to easily vary its stiffness and damping. When optimization was used to tune the stance-knee stiffness of the regular-walking, step-down, and step-up controllers, a pattern similar to that observed in humans was found: a virtual compliant element for the step-down controller is stiffer than for the regular walking controller, and a softer virtual compliant element was found for the step-up controller.

The tripping-reflex controller is designed to behave in different manners with respect to early or late tripping: A rapid elevating strategy of the swing leg is activated in response to early tripping while a rapid lowering strategy is used in response to late tripping. This rapid lowering strategy is followed by a recovery phase which is designed to reject the previous step's disturbance.

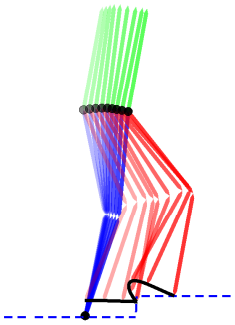
Experimentation with this controller showed MABEL traversing various kinds of obstacles including 12.5 cm blind step-ups, and 18.5 cm blind step-downs, as well as various combinations of step-ups and step-downs. Limitations on the results will be discussed in Chapter VIII.



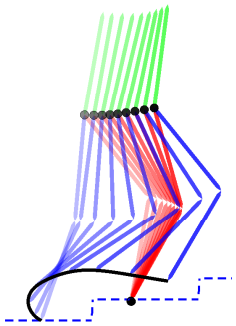
(a) RW → RE → SU → RE → RC → SU



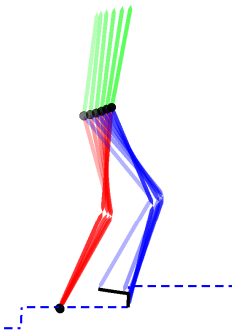
(b) RW



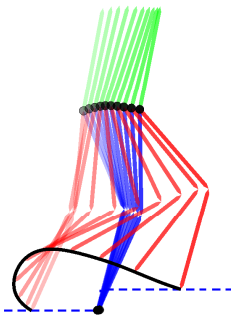
(c) RW → RE



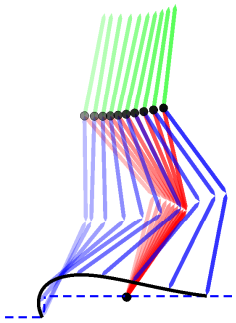
(d) SU



(e) SU → RE

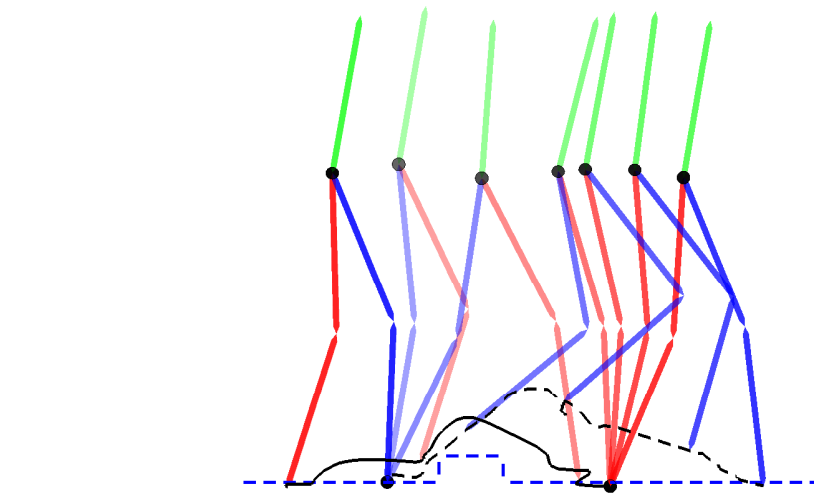


(f) RC

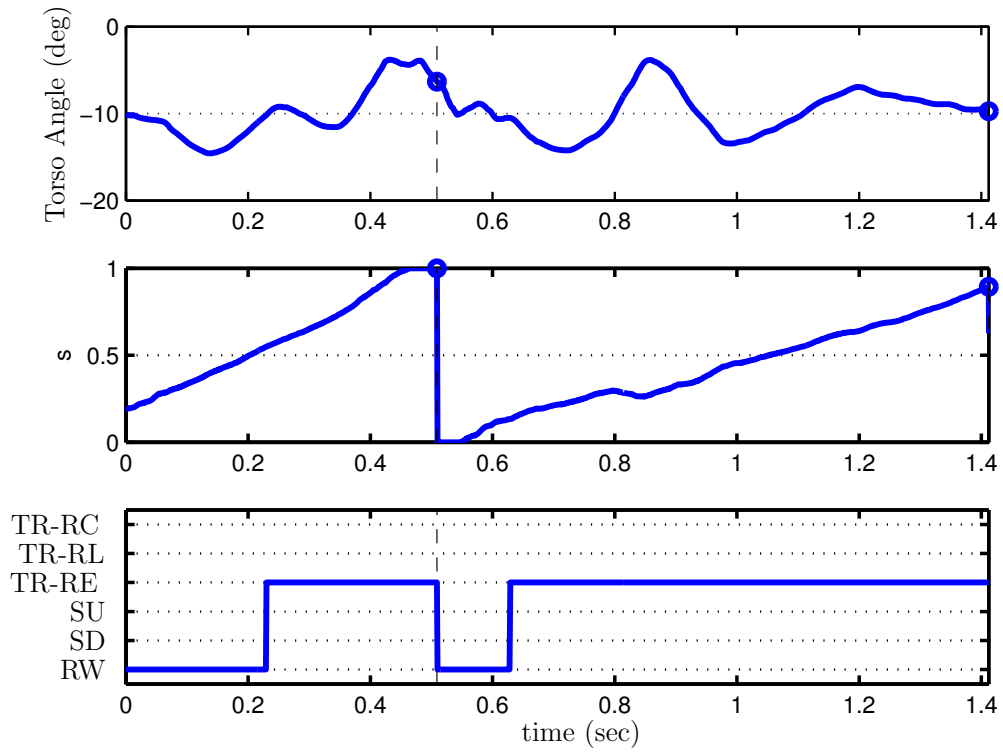


(g) SU

Figure 7.5: Example of simulated sequential composition of controllers. Black circle at the toe indicates stance toe. Robot climbs two unknown stair-steps in this simulation. Black lines indicate swing toe trajectories. (b) Robot starts with regular-walking (c), and has tripped over the first stair, so transition to rapid-elevation occurs. (d) Robot safely places swing toe on top of the first stair, and (e) step-up controller is applied to climb up the first stair. (e) Robot trips over the second stair, and transition to rapid-elevation takes place again, but fails to clear the swing toe at this time. (f) Therefore, recovery controller is applied in order to recover from tripping and safely places swing toe on top of the second stair, (g), and then, step-up controller is activated in response to the second stair-step.



(a)



(b)

Figure 7.6: Experimental data of walking over a 7 cm bump. (a) Stick figure illustration of the experimental data with a black circle indicating the stance toe. (b) Torso angle (top), s (middle), and types of controllers (bottom) applied are shown, respectively. Blue hollow circle and vertical gray dashed line mark the end of the step.

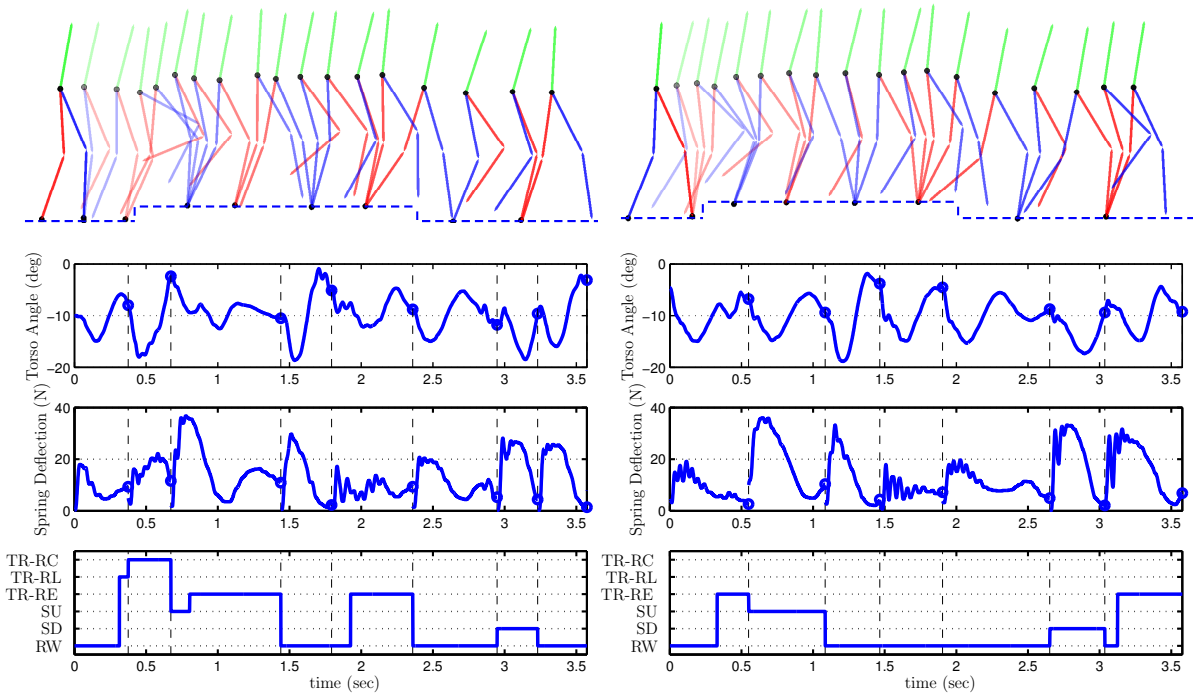


Figure 7.7: Experimental data of step-up and step-down of 10.5 cm platform. In stick figure illustration, black circle at the toe indicates stance toe. Torso angle (top), s (middle), and types of controllers (bottom) applied are shown next, respectively. In these figures, blue hollow circle and black dashed line indicate the end of the step. RW, SD, SU, TR-RE, TR-RL, TR-RC refer walking on flat ground, step-down, step-up, rapid-elevation of the swing leg, rapid-lowering of the swing, and recovery, respectively. (a) First Lap (b) Second Lap

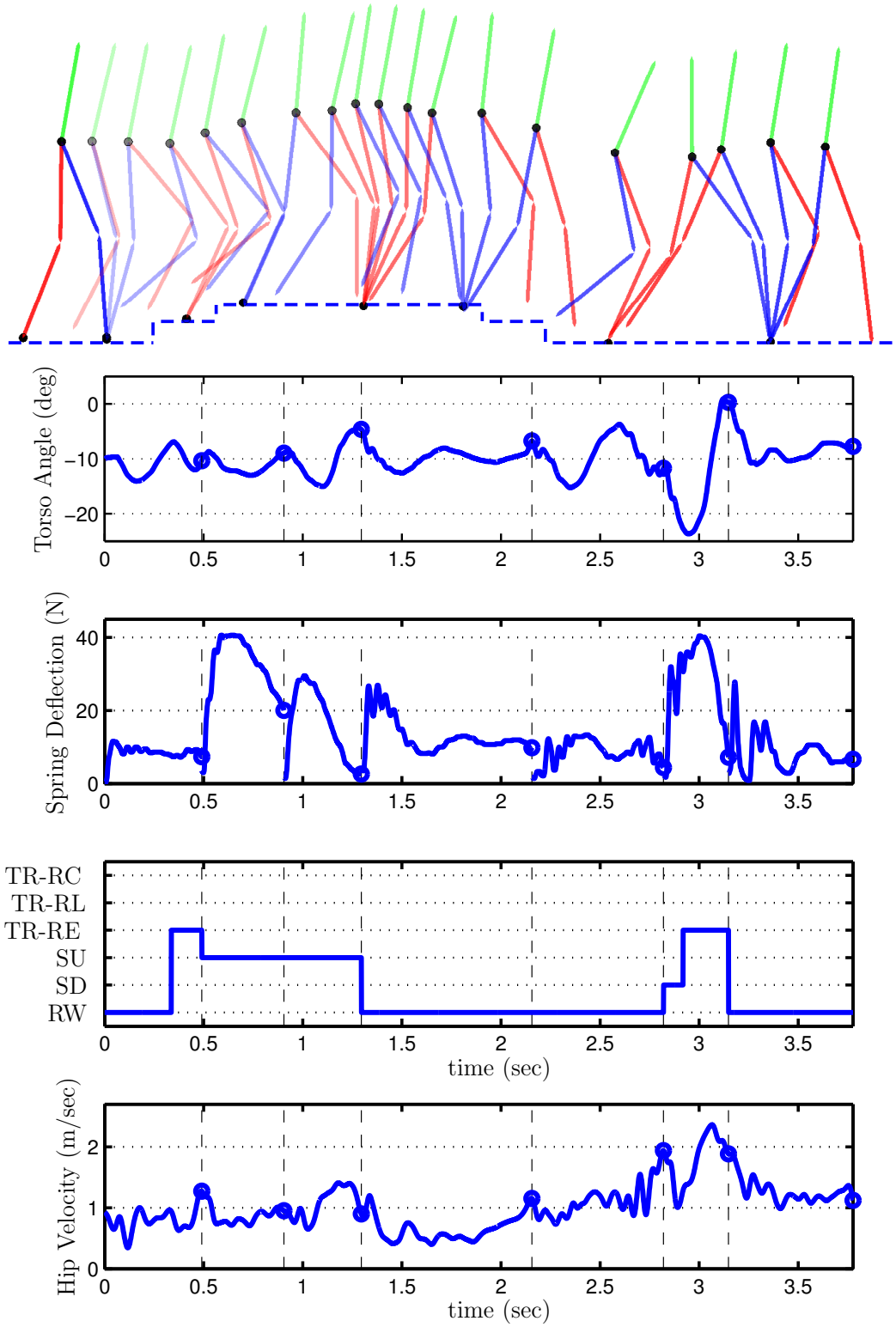


Figure 7.8: Experimental data of consecutive two step-up and one large step-down of 18.5 cm platform. (a) Stick figure illustration. Black circle at the toe indicates stance toe. (b) Torso angle, spring deflection, types of controllers applied, and horizontal hip velocity are shown, respectively. Blue hollow circle and black dashed line display end of the step.

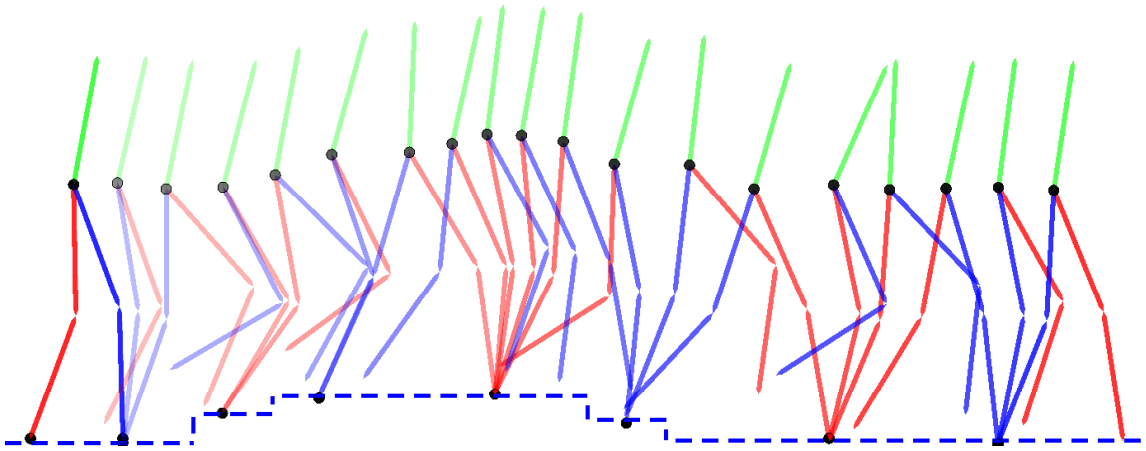


Figure 7.9: Experimental data of two step-ups and two step-downs. Black circle at the toe indicates stance toe.

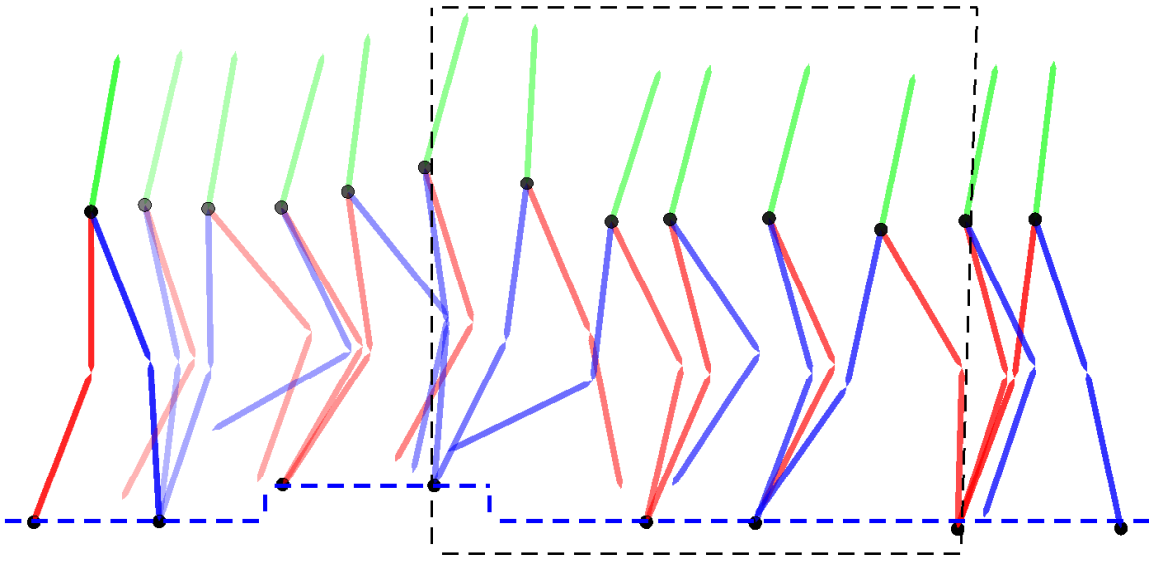


Figure 7.10: Experimental data of one step-up, one regular-walking step, and one step-down of 10.5 cm platform. Steps in black dashed square is redisplayed in Figure 7.11.

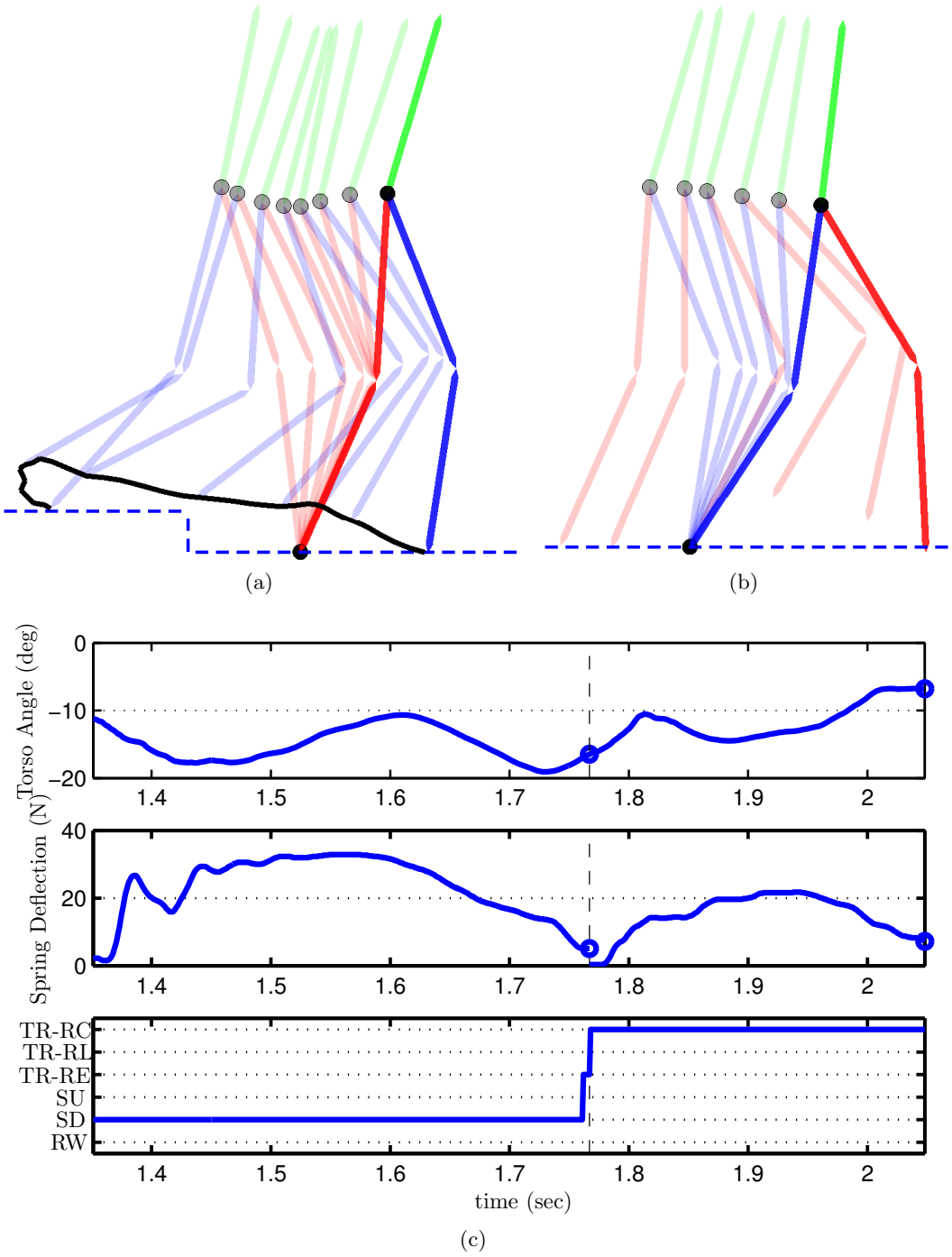


Figure 7.11: Experimental data of one step-up, one regular-walking step, and one step-down of 10.5 cm platform (step-down part). In this figure, only step-down and ensuing step are shown. Stick figure illustration of (a) step-down, and (b) ensuing step. Black circle at the toe and black solid line indicate stance toe and a trajectory of swing toe, respectively. Premature impact with the ground is observed during the step-down. (c) Torso angle, spring deflection, and types of controllers used in the experiments. Blue hollow circle and vertical dashed black line illustrate at the end of the step.

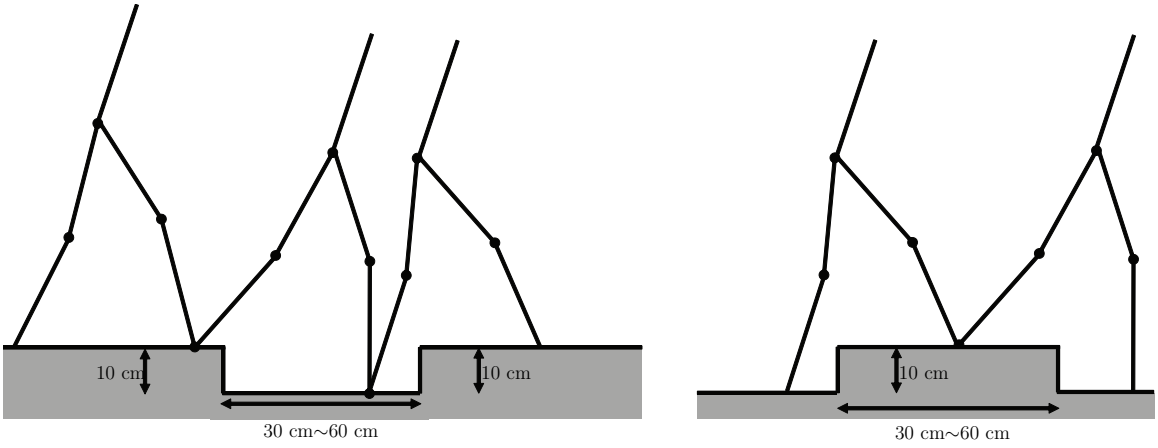


Figure 7.12: Typical examples of terrain causing the robot to fall.

CHAPTER VIII

Concluding Remarks

8.1 Summary of New Contributions

The key results of the thesis are summarized.

(a) *Precise Mathematical Model for Controller Verification*: The six degree of freedom mathematical model which was derived in [100] does not fully reflect experimental reality. As a result, controllers designed on the basis of the model have resulted in walking speeds that are faster than predicted by the dynamic model. Also, the experimentally realized cost of mechanical transport value is approximately three times the designed value [100]. To close the gap between the model and experimental reality, a more detailed model was developed in Chapter III and identified in Chapter IV. The more detailed model includes: (a) a compliant ground model that includes a sophisticated friction model; (b) stretchy cables; and (c) boom dynamics. As depicted in Figure 8.1, the identification process was modular and began with the cable-driven transmission mechanism of the robot. A complete dynamic model of the robot was constructed using the parameters identified in the modular process. The overall model of the robot was validated through a hopping experiment that excited all of the dynamics of the model. The developed and identified detailed model was able to predict dynamics of a set of the hopping experiments with great accuracy. Since its development, all of the control designs implemented on the robot have been verified first on the detailed model. Subsequent experimental results have largely corresponded to those predicted by the model. For example, in this thesis work, the identified model was extensively

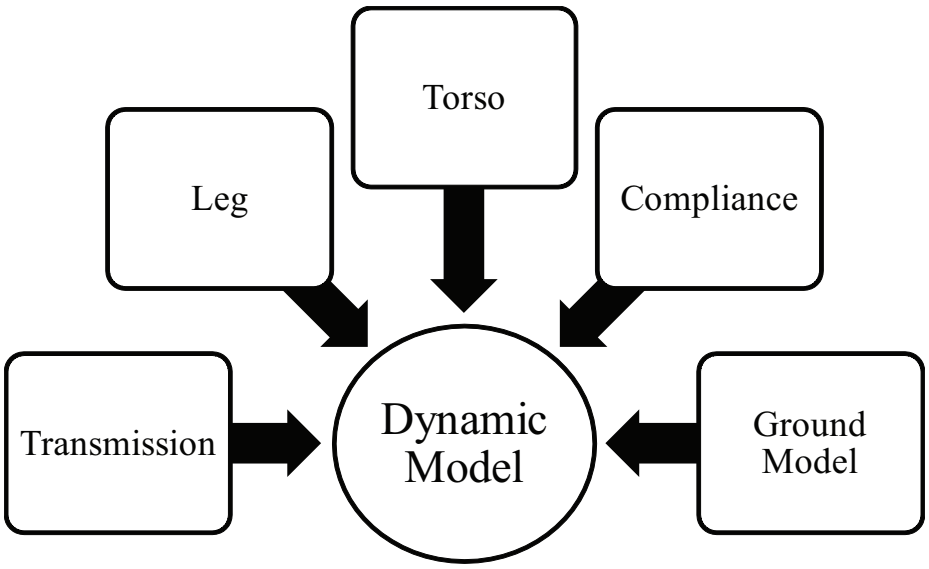


Figure 8.1: System identification process.

used in Chapters V, VI, and VII to verify performance of controllers for stepping-down, stepping-up, and a trip-reflex, and a finite-state machine managing transitions among them.

(b) *Traversing Large Unexpected Ground Variations*: A feedback controller that allows MABEL to traverse terrain that presents large unexpected increases and decreases in height was designed. The robot was provided information on neither where the change in terrain height occurs, nor by how much. Before proceeding with the controller design for large unexpected ground variations, the robustness of the baseline controller was evaluated with a preliminary step-down test. In the experiment, MABEL was able to accommodate a 5.08 cm step-down disturbance, but fell after stepping off a 6.35 cm platform because the leg broke on the ensuing step. Analysis on the experimental data revealed that excessive oscillation of the torso caused the swing leg to strike the ground with sufficient force to break the leg. Therefore, in Chapter V, a specific controller was designed to attenuate torso oscillation after stepping-down. A new controller, called active force control, was introduced which uses the torque input of the leg shape motor to create an additional virtual compliant element by feedback, and optimization was then employed to obtain controller design parameters. The controller was activated when calculated swing foot height at impact was less than some threshold value. Im-

plementation of the controller allowed MABEL to step off an 20.32 cm platform without falling. Drawing on this remarkable experimental result, controllers for stepping-up and a trip reflex were also sought in Chapter VI. In Chapter VII, a finite-state machine was designed that managed transitions among controllers for flat-ground walking, stepping-up, stepping-down, and a trip reflex. If the robot completes a step, the depth of a step-down or height of a step-up would be immediately estimated at impact from the lengths of the legs and the angles of the robot's joints. The change in height was used to invoke a proper control response. On the other hand, if the swing leg impacted an obstacle during a step, or had a premature impact with the ground, a trip reflex was triggered on the basis of specially designed contact switches (see Figure A.1c) on the robot's shins, contact switches on the end of each leg, and the current configuration of the robot. The designed controller yielded experimental results of MABEL (blindly) accommodating various types of platforms, including ascent of a 12.5 cm high stair, stepping-off an 18.5 cm high platform, and walking over a platform with multiple ascending and descending steps. These experimental results surpassed one of the requirements of the W-Prize (see Chapter I), specifically, the robustness-test called the "now-you-see-it-now-you-don't ditch", that no other bipedal robot has satisfied.

8.2 Perspectives on Future Work

Although the proposed finite-state machine shows good performance on various terrain profiles, the following issues still remain unresolved.

(a) *Stability Analysis* A stability analysis on the finite-state machine was not carried out in this thesis. In fact, it is not even clear what "stability" means in this context. Because of non periodicity of gaits induced by the variations in ground height, a Poincaré map cannot be constructed for the closed-loop system. As considered in [84], a Poincaré map can be defined for specific periodic sequences of events, and stability of the associated periodic orbit then determined by the Poincaré map. However, the stability analysis based on the constructed Poincaré map is only valid for those specific sequences and exactly identical terrain profiles. Unfortunately, small changes in the terrain profiles

can result in a totally different sequence of controllers, which breaks the Poincaré map construction. Therefore, it is impossible to analyze all of the possible combinations of phases and terrain profiles via Poincaré maps.

One way to design a set of controllers equipped with a finite-state machine with guaranteed stability was shown in [124]. However, the method requires two things: (a) each controller induces to an asymptotically stable periodic orbit, and (b) the computation of the basin of attraction for each controller, which takes a tremendous amount of time for a complex system. Although many computational approaches for estimating the basin of attraction of nonlinear systems were introduced recently in [107, 108, 64], they still lack practicality for a complex large degree-of-freedom system to the best knowledge of the author.

(b) *Analytical Selection of Switching Surface* The switching surface plays an important role in the proposed controller because, based on the switching surface, switching from one controller to another controller occurs. For the proposed controllers, the switching surfaces were selected by hand tuning of threshold functions. For example, the step-up or step-down controller is activated when the calculated height of the swing toe at the impact is greater or less than the hand-tuned value, respectively. The tripping-reflex controller is also activated by comparing threshold functions which are a function of $s(q)$ or the absolute swing leg angle to hand-tuned values (see (6.15) and (6.5) in Section 6.2). Instead of obtaining these threshold values by hand-tuning, the values can be obtained in a more analytical way, such as an iterative simulation study or optimization technique. Furthermore, the threshold-values of the scalar function of the robot's state could be outcomes of the finite-state machine for more adaptability to the states of the robot or the environment.

(c) *Analytical Selection of Virtual Compliance* The set of parameters of the virtual compliance for each controllers is locally modified with respect to the additional bend in the stance knee as discussed in Section 7.3. It would be interesting to explore a more systematic modification of the virtual compliance through event-based LQR design as in [81, 99].

(d) *Design of Swing Leg Control*

In order to avoid the leg scuffing the ground as well as provide a good beginning pose for the stance leg at the ensuing step, the swing leg profile should be planned carefully. In the proposed controller, virtual constraints for the swing leg are imposed on the motor leg shape and leg angle. Therefore, if there is an additional bend at the stance knee or excessive torso pitching, then the swing leg may not have enough leg clearance to finish the step as planned. As a partial solution to this problem, we can impose virtual constraints to the vertical and horizontal position of the swing toe, respectively. Then swing leg vertical and horizontal position should be regulated even though there is an additional bend at the stance knee or excessive torso pitching. However, this is only a partial solution for the following reasons. Firstly, if torso tracking is poor and there is a great deal of oscillation (this large torso oscillation is observed in baseline controller step-down experiment as discussed in Section 5.1.), this large oscillating error will be translated to the swing leg control because the position of the swing toe is also a function of torso angle. Therefore, this oscillating error can cause oscillation in the torques in the actuators for the swing leg. Secondly, if the desired swing toe position is outside of the workspace of the swing leg, then singularity occurs.

(e) *Sequential Composition of Local Feedback Policies* The proposed controller can be considered a sequential composition of planned motions which are obtained through offline optimization, taking the robot's state into the region of attraction of the regular-walking controller in response to a large deviation due to the disturbance. One candidates to solve this kind of problem can be sequential composition of local feedback policies as suggested in [9, 108]. We can define sets of the deviated robot's state due to various kinds of typical disturbances, including ground variations in height, slipping, and tripping over obstacles. For the defined sets of state deviations, we can build a sequence of local feedback controllers which steer the state from the defined sets to the region of attraction of the regular-walking controller. However, this technique is computationally unpleasant, especially for a large degree-of-freedom complex nonlinear system, and sensitive to sensor noise or delay when experimentally implemented [94]. Furthermore,

the issue of how to choose subsets of the robot's state which represent disturbed states due to the various kinds of disturbances is not a trivial problem.

(f) *Robust Control Design* The video in [75] shows limited controller performance to attenuate the influence of the ground height variations on the robot's speed. To improve controller performance in the face of such disturbances, an outer-loop discrete-event-based control design can be layered onto the baseline controller. Because variations in ground height affect the robot stride-to-stride, the controller design problem for walking over uneven ground can be viewed as a disturbance attenuation problem of a discrete system if we update the parameters in the controller stride-to-stride. The following robust optimal control problem can be posed to solve such a problem: design a discrete update law of control parameters u_k, such as parameters in the virtual compliance or parameters in the virtual constraints, to bound or minimize certain desired output $y(q_k)$ which is a function of the robot's state, such as, the magnitude of the deviated state from the fixed point or difference in the horizontal average velocity from the fixed point average velocity, in response to an unknown disturbance w_k, that is, variations in ground height. Here, q_k is the robot's state sampled stride-to-stride. Various techniques can be employed to design a robust controller, including robust dynamic programming, robust model predictive control, H^∞ control, and a game theory approach. Appendix H shows preliminary simulation result on the design of robust outer-loop discrete-event-based control design. A robust control design for improving MABEL's walking performance in the presence of persistent random variations in step height is proposed. Modification is made on the torso control. H^∞ methods and Quasilinear Control Theory (QLC) [17] were employed to attenuate the effects of the disturbance with input saturation considered. In simulation, the controller design leads to greater success of walking on randomly varying ground.

APPENDICES

APPENDIX A

Novel Mechanical Design

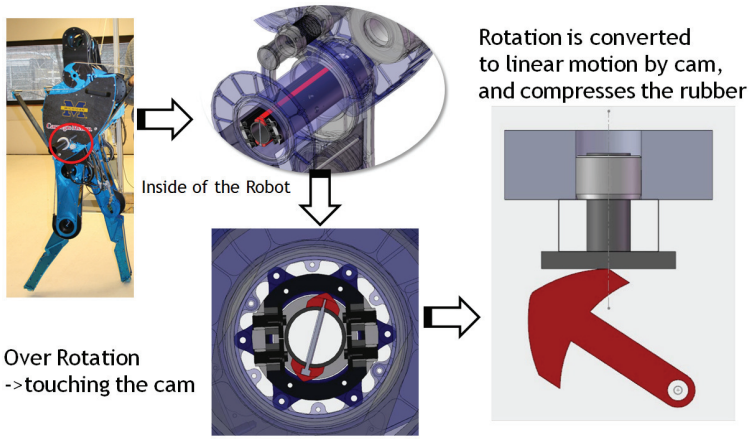
Novel mechanisms which have been designed and installed on MABEL in the course of this thesis work are summarized here.

Novel Design of Rotation Limiter

For mechanical protection, a rotation limiter for MABEL's torso was designed, fabricated, and implemented. The motors and most of the pulleys forming the differentials are inside the torso, so the robot's center of mass is above the hip. Therefore, the robot tends to fall head first. Consequently, if the control activity fails to stabilize the robot, the torso impacts the ground which could potentially damage many parts. To prevent this, torso rotation needs to be limited over some specified range. The rotation limiter has a mechanism which behaves like a cam and a cam follower to convert rotational impact to linear impact. The converted linear impact is then absorbed by compressing a rubber-based material, and gently stops the rotation of the robot beyond the designed range. Figure A.1a shows where the rotation limiter is installed and how it works.

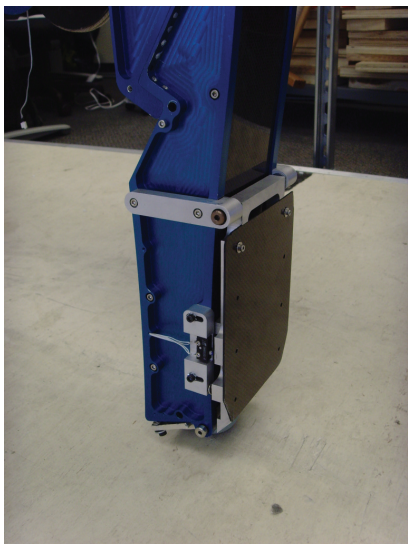
Passive Feet

Hardware modifications were carried out on MABEL to replace the shins on both legs with a different pair of shins that terminate in passive feet enclosed in a pair of regular running sneakers (see Figure A.1b). After a minimal modification to the nominal walking



(a)

(b)



(c)

Figure A.1: Novel Mechanical Design (a) Rotation Limiter (b) Passive feet (c) Switch detecting impacts with obstacles.

controller, MABEL walked with passive feet, exhibiting the classic heel-strike and toe-off typical of human walking. The video is available at [76].

APPENDIX B

How to Estimate Motor-Torque Bias

In normal operation of the robot, the commanded motor torques may vary from -10 to 10 N-m or higher, in which case a bias of 0.1 N-m in the commanded torque is insignificant. We are using the pulleys comprising the differentials as a known load when identifying the motor characteristics. Because the pulleys have low inertia as shown in Table 4.2, the commanded motor torques are approximately 1 N-m, in which case a torque bias of 0.1 N-m is significant.

For each parameter estimation experiment, the motor-torque biases are estimated and removed by the following procedure. First, in order to minimize the effect of friction from the remaining pulleys, each motor pulley is isolated by disconnecting the cable between the motor and the rest of the transmission. Each motor is actuated with a zero-mean sinusoidal torque command. An amplifier bias is expected to cause the motor position to drift slowly, as shown in Figure B.1. Differentiating the measured motor position gives the angular velocity, as shown in Figure B.2.

The transfer function between the torque input and the angular velocity can be modeled as a first-order system, similar to (4.1), with an additional step input. Identification of the bias is therefore accomplished with the two-input first-order autoregressive model [62, pp. 71-73]

$$y_k = a_1 y_{k-1} + b_1 u_{1,k-1} + b_2 u_{2,k-1}, \tag{B.1}$$

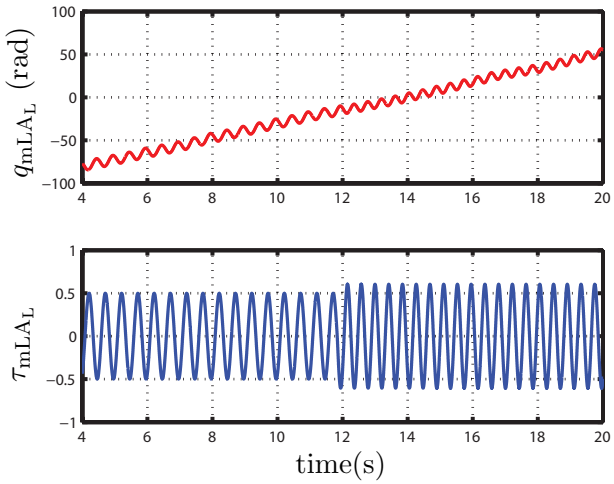


Figure B.1: Amplifier bias. A zero-mean command input to the amplifier produces a motor position output that slowly drifts with time, showing that the amplifier is biased. The drift is independent of the input amplitude and frequency.

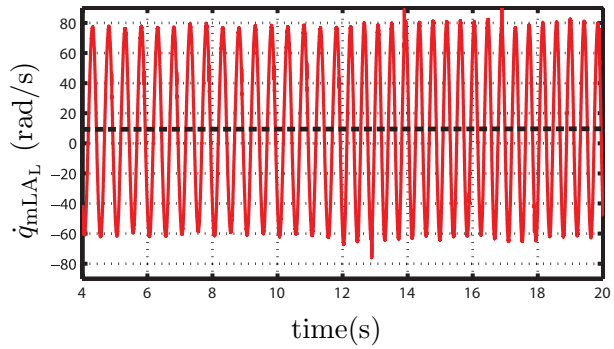


Figure B.2: Differentiated output. The mean value of the differentiated signal, which is given by the dashed line, is nonzero.

where y is the motor angular velocity, u_1 is the commanded motor torque, and u_2 is a sequence of 1's. Rearranging (B.1) gives

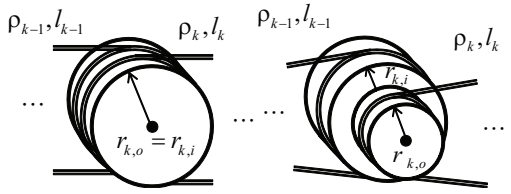
$$y_k = a_1 y_{k-1} + b_1(u_{1,k-1} + b_2/b_1). \tag{B.2}$$

The bias is removed by subtracting b_2/b_1 from the commanded motor torque.

APPENDIX C

Inertia Lumping

Consider the k^{th} pulley in a serial connection of the pulleys shown in Figure C.1. The input and output radius of the k^{th} pulley are denoted by $r_{k,i}$ and $r_{k,o}$, respectively, as shown in Figure C.1. While the input and output radius of a standard pulley are identical, the input and output radius of a step-down pulley are different.



(a) Standard Pulley (b) Step-down Pulley

Figure C.1: Types of pulleys. (a) Standard pulley: input radius and output radius are identical. (b) Step-down pulley: input radius and output radius are different, and the ratio of the radii is the gear ratio.

The moment of inertia of the k^{th} pulley seen from the $k-1^{th}$ pulley is obtained by

$$J_k^{k-1} = \frac{r_{k-1,o}^2}{r_{k,i}^2} J_k, \tag{C.1}$$

where J_k is the k^{th} pulley's moment of inertia. The k^{th} cable inertia seen from the $k-1^{th}$ cable can also be derived:

$$J_{cable,k}^{k-1} = \frac{r_{k-1,o}^2}{r_{k,i}^2} J_{cable,k} \tag{C.2}$$

$$J_{cable,k} = r_{k,i}^2 \rho_k l_k, \tag{C.3}$$

where ρ_k is the density of the k^{th} cable, and l_k is the length of the k^{th} cable. By applying (C.1) and (C.2) consecutively from k^{th} to 0^{th}, the k^{th} pulley and cable inertia as seen from the 0^{th} pulley (the motor pulley) can be obtained as follows:

$$J_k^0 = \frac{r_{0,o}^2}{r_{1,i}^2} \cdots \frac{r_{k-2,o}^2}{r_{k-1,i}^2} \frac{r_{k-1,o}^2}{r_{k,i}^2} J_k^k, \tag{C.4}$$

$$J_{cable,k}^0 = \frac{r_{0,o}^2}{r_{1,i}^2} \cdots \frac{r_{k-2,o}^2}{r_{k-1,i}^2} \frac{r_{k-1,o}^2}{r_{k,i}^2} J_{cable,k}^k. \tag{C.5}$$

The lumped moment of inertia of the pulley combination is obtained by summing up all of the pulley and cable moments of inertia as viewed from the 0^{th} pulley:

$$J_{pulley} = \sum_{k=1}^{N} J_k^0 = \sum_{k=1}^{N} \frac{r_{0,o}^2}{r_{1,i}^2} \cdots \frac{r_{k-2,o}^2}{r_{k-1,i}^2} \frac{r_{k-1,o}^2}{r_{k,i}^2} J_k^k \tag{C.6}$$

$$J_{cable} = \sum_{k=1}^{N-1} J_{cable,k}^0 = \sum_{k=1}^{N-1} \frac{r_{0,o}^2}{r_{1,i}^2} \cdots \frac{r_{k-2,o}^2}{r_{k-1,i}^2} \frac{r_{k-1,o}^2}{r_{k,i}^2} J_{cable,k}^k, \tag{C.7}$$

where N is the total number of the pulleys. This lumped moment of inertia of the pulley combination is substituted into (4.1), which gives the transfer function from the motor torque to the motor angle.

APPENDIX D

Details of Hopping Controller

A hopping gait is used to complete the identification of, and subsequently validate, the overall dynamic model of MABEL. A heuristic controller for two-legged hopping is formed by decomposing a hopping gait into its elementary phases, as illustrated in Figure D.1. For each phase $\mathrm{X} \in \{\mathrm{I}, \mathrm{II}, \mathrm{III}, \mathrm{IV_a}, \mathrm{IV_b}, \mathrm{V}\}$ of Figure D.1, the controlled variables are

$$h := \begin{bmatrix} q_{\mathrm{LA_L}} \\ q_{\mathrm{mLS_L}} \\ q_{\mathrm{LA_R}} \\ q_{\mathrm{mLS_R}} \end{bmatrix}, \tag{D.1}$$

while the reference command h_{X}^{ref} changes in each phase as given in (D.3)-(D.15) below. The PD controller

$$u = K_{\mathrm{p}} \left(h_{\mathrm{X}}^{ref} - h \right) + K_{\mathrm{d}} \left(-\dot{h} \right) \tag{D.2}$$

is used, where K_{p} is a 4×4 diagonal matrix of proportional gains and K_{d} is a 4×4 diagonal matrix of derivative gains. In the simulation model, h in (D.2) is quantized to the same level as the encoders on the robot, and $\dot{h}$ is obtained by numerical differentiation. The control inputs are updated with a sampling time of 1 ms, which equals the sampling time used on the robot.

Phase I, Flight

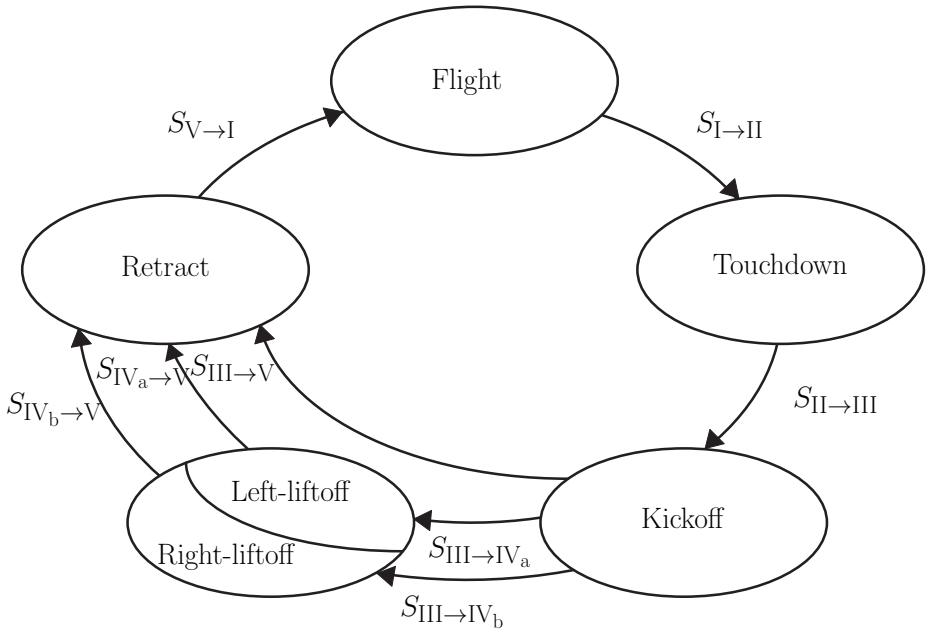


Figure D.1: Hopping controller phases and transitions. From phase III, three possible transitions can occur because which leg first comes off the ground cannot be predicted. According to which leg comes off the ground, the controller selects phase $\mathrm{IV_a}$, $\mathrm{IV_b}$, or V as the next phase.

The flight phase is characterized by the absence of contact with the ground. Because the robot is in the air and there is nothing to push against, the reference commands

$$h_{\mathrm{I}}^{ref} = \begin{bmatrix} \pi - 0.5\delta_{\mathrm{LA}} - h^d_{\mathrm{Tor}} \\ \gamma_{\mathrm{LS}\to\mathrm{mLS}} h^d_{\mathrm{LS_L}} \\ \pi + 0.5\delta_{\mathrm{LA}} - h^d_{\mathrm{Tor}} \\ \gamma_{\mathrm{LS}\to\mathrm{mLS}} h^d_{\mathrm{LS_R}} \end{bmatrix}, \tag{D.3}$$

$$S_{\mathrm{I}\to\mathrm{II}} : \{p_{\mathrm{toe_L}} = 0, p_{\mathrm{toe_R}} = 0\}, \tag{D.4}$$

focus on the robot's relative pose rather than the absolute orientation of a link with respect to a world frame. While the flight-phase controller runs until both legs are in contact with the ground, the time between one leg and the other impacting the ground is less than 20 ms, and hence for most of this phase the robot is in the air.

Constant setpoints are chosen for the controlled variables (D.1). Specifically, the commands $q_{\mathrm{LA_L}} = \pi - 0.5\delta_{\mathrm{LA}} - h^d_{\mathrm{Tor}}$ and $q_{\mathrm{LA_R}} = \pi + 0.5\delta_{\mathrm{LA}} - h^d_{\mathrm{Tor}}$ regulate the relative angle between the legs to a desired value of δ_{LA} and orient the legs with respect to a nominal torso

angle of h^d_{Tor}. The commands $q_{\text{mLS}_\text{L}} = \gamma_{\text{LS}\to\text{mLS}} h^d_{\text{LS}_\text{L}}$ and $q_{\text{mLS}_\text{R}} = \gamma_{\text{LS}\to\text{mLS}} h^d_{\text{LS}_\text{R}}$ regulate the lengths of the left and right legs to nominal values $h^d_{\text{LS}_\text{L}}$ and $h^d_{\text{LS}_\text{R}}$, respectively; equivalently, these commands can be thought of as setting the knee angles at appropriate values for absorbing the subsequent impact. The gear ratios $\gamma_{\text{LS}\to\text{mLS}}$ and $\gamma_{\text{LS}\to\text{mLS}}$ are present because the controlled variables are motor positions.

Phase II**, Touchdown**

With both legs firmly on the ground, the reference command becomes

$$h^{ref}_{\text{II}} = \begin{bmatrix} q_{\text{Tor}} + q_{\text{LA}_\text{R}} - h^d_{\text{Tor}} - \delta_{\text{LA}} \\ \gamma_{\text{LS}\to\text{mLS}} h^d_{\text{LS}_\text{L}} \\ q_{\text{Tor}} + q_{\text{LA}_\text{R}} - h^d_{\text{Tor}} \\ \gamma_{\text{LS}\to\text{mLS}} h^d_{\text{LS}_\text{R}} \end{bmatrix}, \tag{D.5}$$

$$S_{\text{II}\to\text{III}} : \{|\dot{p}^v_{\text{hip}}| < 0.0225\}. \tag{D.6}$$

The leg-angle motors are now used to regulate the absolute orientation of the torso to h^d_{Tor} and the relative angle between the legs to δ_{LA}. The leg-shape motor positions q_{mLS_L} and q_{mLS_R} continue to be commanded to constant values so that the springs absorb the impact energy. The transition to the kickoff phase occurs when the vertical velocity of the center of mass approaches zero.

Phase III**, Kickoff**

When the vertical component of the center of mass velocity approaches zero, the reference command is changed to

$$h^{ref}_{\text{III}} = \begin{bmatrix} q_{\text{Tor}} + q_{\text{LA}_\text{R}} - h^d_{\text{Tor}} - \delta_{\text{LA}} \\ \gamma_{\text{LS}\to\text{mLS}} h^d_{\text{LS}_\text{L}} - \gamma_{\text{LS}\to\text{mLS}} \delta^-_{\text{LS}_\text{L}} \\ q_{\text{Tor}} + q_{\text{LA}_\text{R}} - h^d_{\text{Tor}} \\ \gamma_{\text{LS}\to\text{mLS}} h^d_{\text{LS}_\text{R}} - \gamma_{\text{LS}\to\text{mLS}} \delta^-_{\text{LS}_\text{R}} \end{bmatrix}, \tag{D.7}$$

$$S_{\text{III}\to\text{IV}_\text{a}} : \{p_{\text{toe}_\text{L}} > 0, p_{\text{toe}_\text{R}} \leq 0\}, \tag{D.8}$$

$$S_{\text{III}\to\text{IV}_\text{b}} : \{p_{\text{toe}_\text{L}} \leq 0, p_{\text{toe}_\text{R}} > 0\}, \tag{D.9}$$

$$S_{\text{III}\to\text{V}} : \{p_{\text{toe}_\text{L}} > 0, p_{\text{toe}_\text{R}} > 0\}. \tag{D.10}$$

To propel the robot off the ground, the legs are extended by $\delta^{-}_{\mathrm{LS_L}}$ for the left leg and $\delta^{-}_{\mathrm{LS_R}}$ for the right leg. The remaining commands are unchanged. From phase III, three possible transitions can occur because which leg comes off the ground first cannot be predicted, and, with a 1-ms update rate, it is possible that both legs are observed to leave the ground simultaneously. According to leg liftoff order, the controller chooses phase $\mathrm{IV_a}$, $\mathrm{IV_b}$, or V as the next phase.

Phase $\mathrm{IV_a}$, Left-liftoff

When the left leg lifts off the ground while the right leg remains in contact with the ground, the reference command becomes

$$h^{ref}_{\mathrm{IVa}} = \begin{bmatrix} H_1 h^{ref}_{\mathrm{III}}\left(t^{*}_{\mathrm{III}}\right) \\ \gamma_{\mathrm{LS}\to\mathrm{mLS}} h^{d}_{\mathrm{LS_L}} + \gamma_{\mathrm{LS}\to\mathrm{mLS}}\delta^{+}_{\mathrm{LS_L}} \\ H_3 h^{ref}_{\mathrm{III}}\left(t^{*}_{\mathrm{III}}\right) \\ \gamma_{\mathrm{LS}\to\mathrm{mLS}} h^{d}_{\mathrm{LS_R}} - \gamma_{\mathrm{LS}\to\mathrm{mLS}}\delta^{-}_{\mathrm{LS_R}} \end{bmatrix}, \tag{D.11}$$

$$S_{\mathrm{IV_a}\to\mathrm{V}} : \{p_{\mathrm{toe_R}} > 0\}, \tag{D.12}$$

where $H_1 = [1\ 0\ 0\ 0]$, $H_3 = [0\ 0\ 1\ 0]$, and t^{*}_{III} is the time when the transition from phase III occurs. The left leg starts to retract by $\delta^{+}_{\mathrm{LS_L}}$ to provide clearance. The leg-angle positions are held at their values from the end of phase III.

Phase $\mathrm{IV_b}$, Right-liftoff

When the right leg lifts off the ground while the left leg remains in contact with the ground, the reference command becomes

$$h^{ref}_{\mathrm{IVb}} = \begin{bmatrix} H_1 h^{ref}_{\mathrm{III}}\left(t^{*}_{\mathrm{III}}\right) \\ \gamma_{\mathrm{LS}\to\mathrm{mLS}} h^{d}_{\mathrm{LS_L}} - \gamma_{\mathrm{LS}\to\mathrm{mLS}}\delta^{-}_{\mathrm{LS_L}} \\ H_3 h^{ref}_{\mathrm{III}}\left(t^{*}_{\mathrm{III}}\right) \\ \gamma_{\mathrm{LS}\to\mathrm{mLS}} h^{d}_{\mathrm{LS_R}} + \gamma_{\mathrm{LS}\to\mathrm{mLS}}\delta^{+}_{\mathrm{LS_R}} \end{bmatrix}, \tag{D.13}$$

$$S_{\mathrm{IV_b}\to\mathrm{V}} : \{p_{\mathrm{toe_L}} > 0\}. \tag{D.14}$$

The notation is as in phase $\mathrm{IV_a}$. The right leg starts to retract by $\delta^{+}_{\mathrm{LS_R}}$ to provide clearance. The leg-angle positions are held at their values from the end of phase III.

Phase V, Retract

Once both legs lift off the ground, the retraction phase is held for 50 ms, after which the controller passes to the flight phase. The reference command is

$$h_{\mathrm{V}}^{ref} = \begin{bmatrix} H_1 h_{\mathrm{III}}^{ref}\left(t_{\mathrm{III}}^*\right) \\ \gamma_{\mathrm{LS}\to\mathrm{mLS}} h_{\mathrm{LS_L}}^d + \gamma_{\mathrm{LS}\to\mathrm{mLS}} \delta_{\mathrm{LS_L}}^+ \\ H_3 h_{\mathrm{III}}^{ref}\left(t_{\mathrm{III}}^*\right) \\ \gamma_{\mathrm{LS}\to\mathrm{mLS}} h_{\mathrm{LS_R}}^d + \gamma_{\mathrm{LS}\to\mathrm{mLS}} \delta_{\mathrm{LS_R}}^+ \end{bmatrix}, \tag{D.15}$$

$$S_{\mathrm{V}\to\mathrm{I}} : \{t = t_{\mathrm{V}}^{\#} + 0.05\}, \tag{D.16}$$

where $t_{\mathrm{V}}^{\#}$ is the time when the transition to phase V occurs. Otherwise, the notation is as in phase $\mathrm{IV_a}$. The leg-angle positions are held at their values from the end of phase III. Both legs are retracted to provide clearance.

Parameter Values

Applying the above controller to the identified model, a periodic solution is found with the controller parameter values

$$\begin{aligned} \delta_{\mathrm{LA}} &= 0.524 \text{ rad } (30 \text{ deg}), \ \mathrm{h_{Tor}^d} = 0.140 \text{ rad } (8 \text{ deg}), \\ h_{\mathrm{LS_L}}^d &= 0.209 \text{ rad } (12 \text{ deg}), \ \mathrm{h_{LS_R}^d} = 0.209 \text{ rad } (12 \text{ deg}), \\ \delta_{\mathrm{LS_L}}^- &= 0.087 \text{ rad } (5 \text{ deg}), \quad \delta_{\mathrm{LS_R}}^- = 0.227 \text{ rad } (13 \text{ deg}), \\ \delta_{\mathrm{LS_L}}^+ &= 0.087 \text{ rad } (5 \text{ deg}), \quad \delta_{\mathrm{LS_R}}^+ = 0.087 \text{ rad } (5 \text{ deg}), \end{aligned}$$

with the robot's horizontal position over one hop translated backward by 0.27 m, which corresponds to $\delta\phi_{\mathrm{h}}^d = -0.273$ rad $= -6.9$ deg of rotation as measured by the encoder on the central tower.

Simulation shows that the periodic motion is unstable. The orbit is stabilized with an event-based controller that updates the commanded torso angle h_{Tor}^d based on the distance

traveled horizontally during one hop, namely,

$$\begin{aligned}
\phi_{\mathrm{h}}[k] &= \phi_{\mathrm{h}}\left(t_{\mathrm{I}\to\mathrm{II}}\right), \\
\delta h^{d}_{\mathrm{Tor}}[k] &= K_{\mathrm{Tor}}(\phi_{\mathrm{h}}[k] - \phi_{\mathrm{h}}[k-1] - \delta\phi^{d}_{\mathrm{h}}), \\
h^{d}_{\mathrm{Tor}}[k] &= h^{d0}_{\mathrm{Tor}} + \delta h^{d}_{\mathrm{Tor}}[k],
\end{aligned}$$

where k is the hopping count, $t_{\mathrm{I}\to\mathrm{II}}$ is the time when the transition from phase I to phase II occurs, and K_{Tor} is a gain. If MABEL travels less than $\delta\phi^{d}_{\mathrm{h}}$ during the previous hop, the torso is leaned backward from the nominal value h^{d0}_{Tor}, and vice versa if the robot travels more than $\delta\phi^{d}_{\mathrm{h}}$.

A few hours of parameter tuning with the simulation model resulted in a stabilizing controller, while days of trial and error in the laboratory were unsuccessful.

APPENDIX E

Modification on Virtual Compliance and Calculation of Effective Stiffness

MABEL has three sources of compliance: physical springs installed on the torso, virtual compliance created by the leg shape motor, and compliance due to cable stretch. Because the cable stretch is not included in the simplified design model, the stiffness of virtual compliance k_{vc}^* designed from optimization on the simplified design model needs to be modified to take into account the effect of the cable stretch. This modification is based on the fact that the cable stretch and the virtual compliance form a series connection, and consequently, we can multiply k_{vc}^* by some correction factor γ_{vc} so that the series connection realizes the desired compliance k_{vc}^*, namely,

$$\frac{1}{k_{\text{vc}}^*} = \frac{1}{\gamma_{\text{vc}} k_{\text{vc}}^*} + \frac{1}{k_{\text{cable}}}, \tag{E.1}$$

where k_{cable} is the spring constant of the cable stretch seen at the motor leg shape angle. For the designed stiffness value of the regular-walking controller $k_{\text{vc}}^{*,\text{RW}} = 1.8$, $\gamma_{\text{vc}} = 2.5$ is obtained from (E.1), and we use this correction factor for the step-up, step-down, and recovery controllers as well as to make the controller structure simpler.

Similarly, effective stiffness k_{eff} which is the series summation of three sources of compliance can be calculated as follows:

$$\frac{1}{k_{\text{eff}}} = \frac{1}{k_{\text{Bsp}}} + \frac{1}{\gamma_{\text{vc}} k^*_{\text{vc}}} + \frac{1}{k_{\text{cable}}}, \tag{E.2}$$

where k_{Bsp} is the spring constant of physical spring, seen at the motor leg shape angle.

APPENDIX F

Swing Leg Modification for Rapid-Elevation of Swing Leg

As discussed in Section 6.2, the swing leg is elevated rapidly when the robot trips over obstacles in early swing. First, the length of the swing leg is decreased by 15 cm by repositioning the motor leg shape angle (see Figure F.1a). In Figure F.1a, χ is the current leg shape angle, and χ' is the leg shape angle corresponding to 15 cm-decreased leg length. χ' can be calculated from the geometric relationship between those two angles, and is given by,

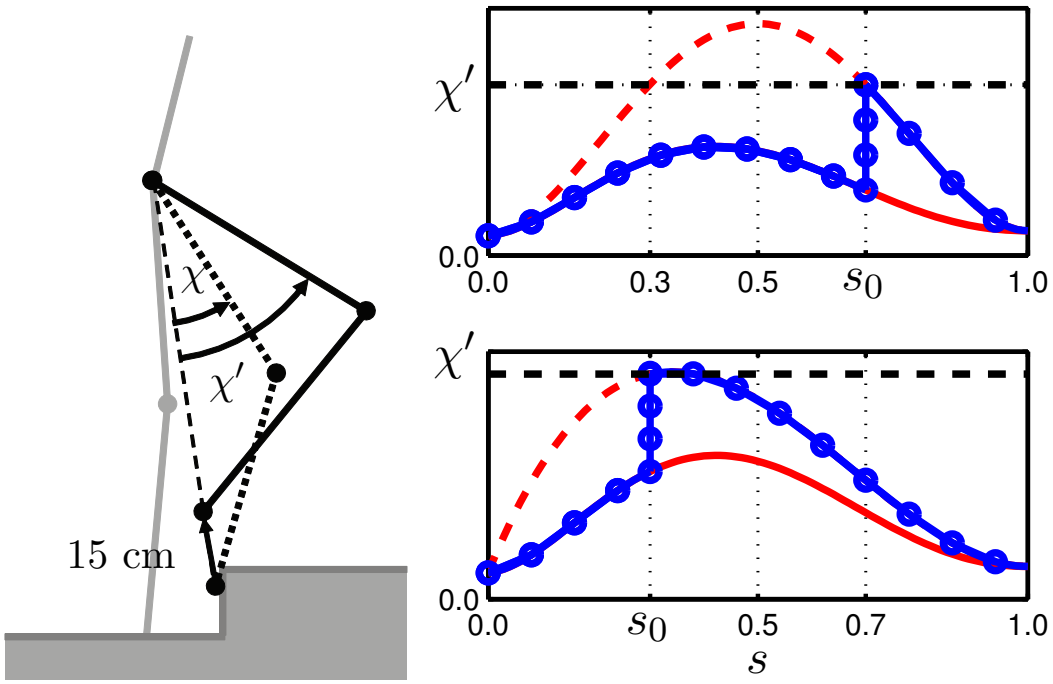
$$\chi' = \arccos\left(\cos(\chi) - 0.15\right) \tag{F.1}$$

Multiplying gear ratio $\gamma_{\mathrm{LS}\to\mathrm{mLS}}$ from LS to mLS to χ' yields the desired motor leg shape angle which is given by,

$$\chi'_{q_{\mathrm{mLS}}} = \gamma_{\mathrm{LS}\to\mathrm{mLS}}\chi' \tag{F.2}$$

Using this desired motor leg shape angle and value of $s(q)$ at the moment of tripping denoted by s_0, we can modify coefficients $\alpha_{\mathrm{mLS_{sw}}}$ of Bézier polynomial $h^d_{\mathrm{mLS_{sw}}}(s, \alpha_{\mathrm{mLS_{sw}}})$ in (5.3) so that the value of modified Bézier polynomial evaluated at s_0 equals to the calculated desired motor leg shape angle, $\chi'_{q_{\mathrm{mLS}}}$.

Because the fifth order Bézier polynomial is used in MABEL, $\alpha_{\mathrm{mLS_{sw}}}$ can be written as $[\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6]$. When $s_0 > 0.5$, we modify only the middle two coefficients, α_3



(a) (b)

Figure F.1: Rapid-elevation of the swing leg. (a) Swing leg is rapidly elevated by repositioning leg shape from χ to χ'. (b) Example of modification on the desired trajectory of the motor leg shape. Blue lines with circle indicate modified desired trajectory for the motor leg shape. Graph on top: when $s(q) > s_0$, Graph on bottom: when $s(q) \leq s_0$

and α_4, by setting two coefficients to the same value $c \in \mathbb{R}$ while the first and last two coefficients are retained (See top graph of Figure F.1b). Whereas, when $s_0 \leq 0.5$, α_2 and α_3 are set to c, and α_4 is set to $0.6c$ so that the swing knee in the middle of gait ($s(q) \approx 0.5$) is not bent too much (See bottom graph of Figure F.1b).

Then, let Bézier coefficient with this modification be $\tilde{\alpha}$. We further define $\hat{\alpha} := [\alpha_1, \alpha_2, 0, 0, \alpha_5, \alpha_6]$ ($\hat{\alpha} := [\alpha_1, 0, 0, 0, \alpha_5, \alpha_6]$ for $s_0 \leq 0.5$) and $\hat{1} := [0, 0, 1, 1, 0, 0]$ ($\hat{1} := [0, 1, 1, 0.6, 0, 0]$ for $s_0 \leq 0.5$). By using $\hat{\alpha}$ and $\hat{1}$, we can rewrite the value of the modified Bézier polynomial evaluated at s_0 which has to be modified to have the same value as $\chi'_{q_{\text{mLS}}}$ as,

$$\begin{aligned} h^d_{\text{mLS}_{\text{sw}}}(s_0, \tilde{\alpha}) &\equiv h^d_{\text{mLS}_{\text{sw}}}(s_0, \hat{\alpha}) + c\, h^d_{\text{mLS}_{\text{sw}}}(s_0, \hat{1}) \\ &= \chi'_{q_{\text{mLS}}}. \end{aligned} \tag{F.3}$$

Rearranging (F.3) gives the value of c

$$c = \frac{\chi'_{q_{\mathrm{mLS}}} - h^d_{\mathrm{mLS_{sw}}}(s_0, \hat{\alpha})}{h^d_{\mathrm{mLS_{sw}}}\left(s_0, \hat{1}\right)}. \tag{F.4}$$

Thus, by setting the middle coefficients α_3 and α_4 to c (α_2 and α_3 to c, and α_4 to $0.6c$ for $s_0 \leq 0.5$) obtained from (F.4), $h^d_{\mathrm{mLS_{sw}}}(s_0, \tilde{\alpha})$ becomes $\chi'_{q_{\mathrm{mLS}}}$. Because last two coefficients kept the same value as $\alpha_{\mathrm{mLS_{sw}}}$, this modified Bézier polynomial will smoothly join with original Bézier polynomial.

Note that the value of $h^d_{\mathrm{mLS_{sw}}}\left(s_0, \hat{1}\right)$ becomes 0 at $s_0 = 0$, thereby, causing infinite value of c. Therefore, in order to avoid this problem, we set $c = 30$ for $s_0 < 0.15$.

APPENDIX G

Screen Captures of the Experimental Videos

Figure G.1: Snapshot: experiment of step-up of 12.5 cm platform. Snapshots are at an interval of 100 msec. The snapshots progress temporally from left to right and from top to bottom.

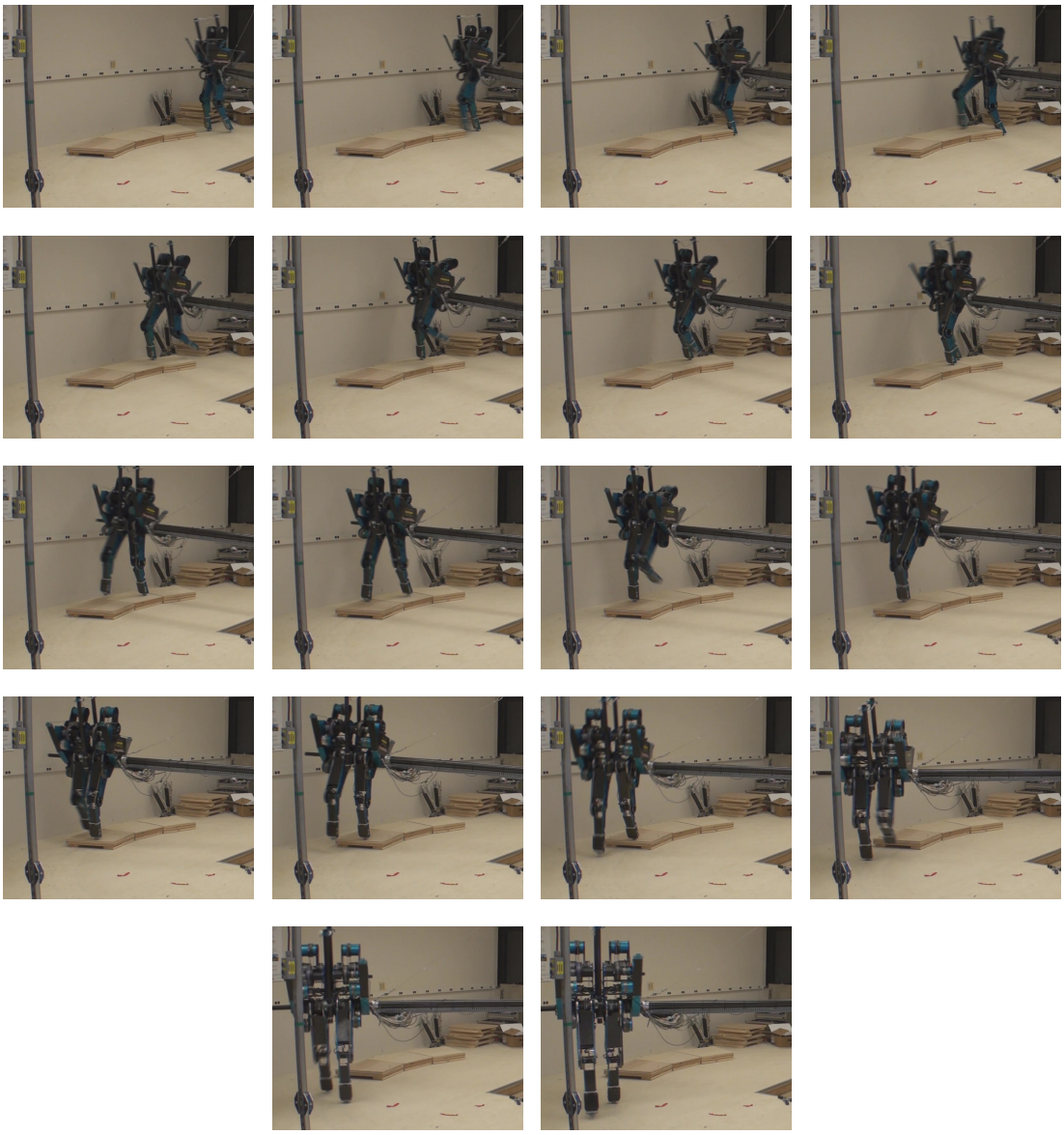


Figure G.2: Snapshot: experiment of step-up and step-down of 10.5 cm platform. Snapshots are at an interval of 200 msec. The snapshots progress temporally from left to right and from top to bottom.

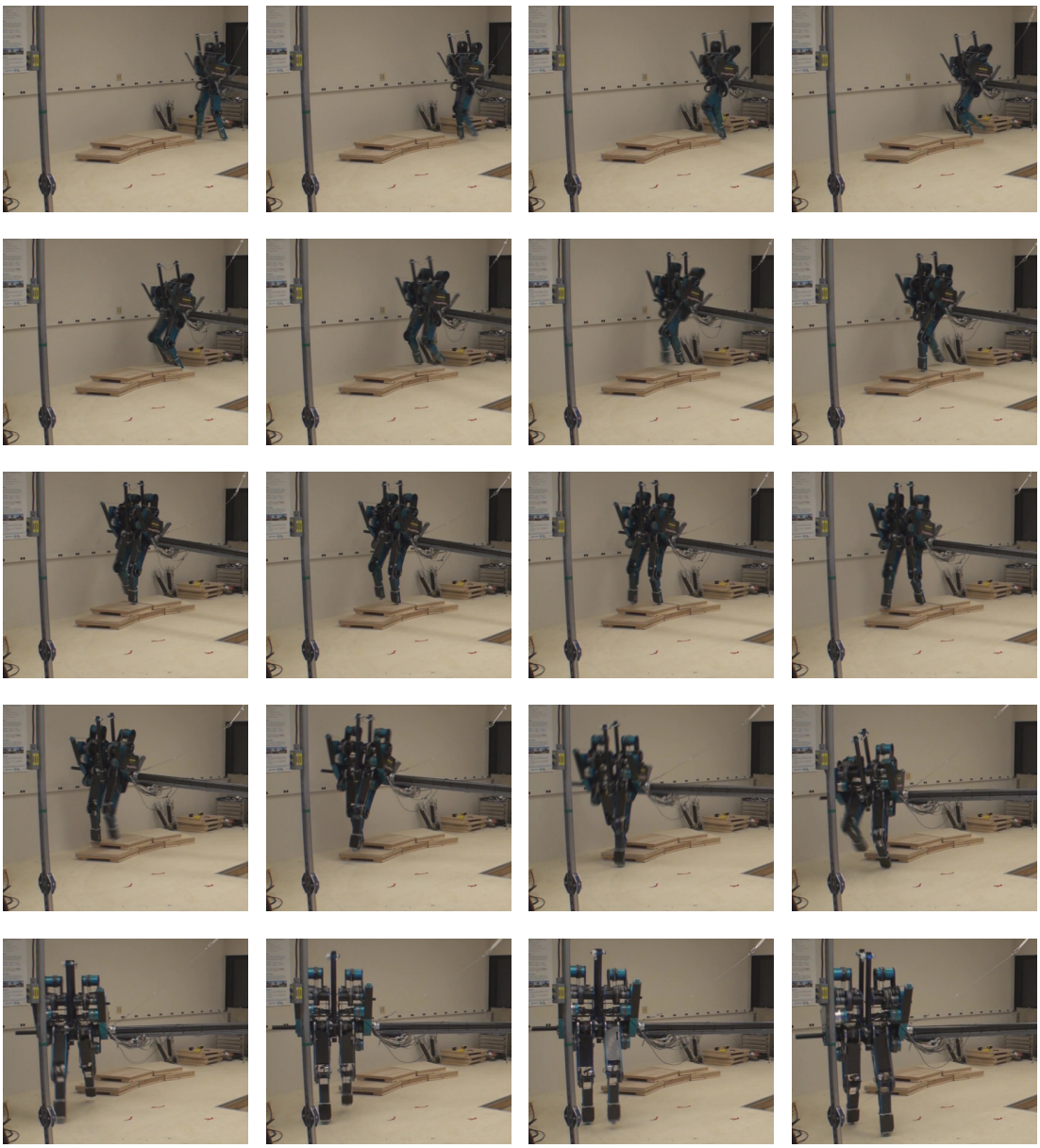


Figure G.3: Snapshot: experiment of two consecutive step-up and one step-down of 18.5 cm platform. Snapshots are at an interval of 200 msec. The snapshots progress temporally from left to right and from top to bottom.

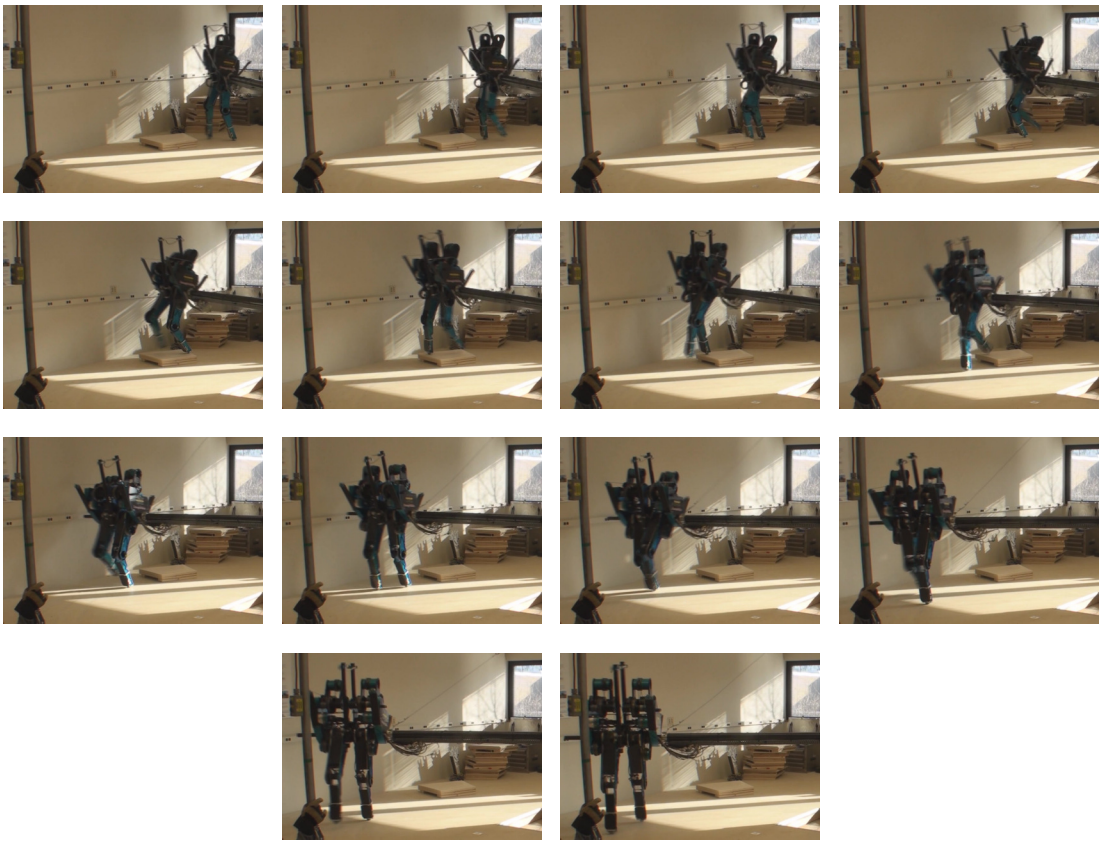


Figure G.4: Snapshot: experiment of step-up, regular-walking, and step-down of 10.5 cm platform. Snapshots are at interval of 200 msec. The snapshots progress temporally from left to right and from top to bottom.

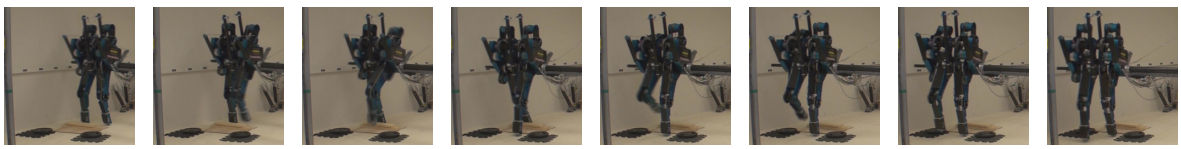


Figure G.5: Snapshot of small bump experiment. Snapshots are at interval of 100 msec. The snapshots progress temporally from left to right and from top to bottom.

APPENDIX H

Control Design for Accommodating Random Variations in Ground Heights

The control designs introduced in Chapter VII allow MABEL to accommodate an abrupt 20 cm decrease and a 12.5 cm increase in ground height. In the experiments conducted with the finite-state machine, MABEL walks over level ground or a slowly increasing ramp for several steps and then walks over a few (2 - 4) obstacles representing considerable decreases and increases ($\geq$ 7 cm) in ground height. After accommodating these obstacles, MABEL again walks over flat ground for several steps or a slowly increasing ramp, and finishes the lap. The ground profiles tested in these experiments can be classified as large occasional disturbances, and the feedback control based on finite-state machine design has shown that it performs well in rejecting these disturbances when the robot has a few steps on level ground in between obstacles.

However, when MABEL walks over continuously randomly varying ground, the controller performance cannot be guaranteed. As discussed in Chapter VIII, the controller has limitations on consecutive disturbances because only one step is considered and optimized in controller design. Further, since the controller has been designed assuming large disturbances, the resulting controller can overreact to small ground variations, and thereby degrade the robot's performance.

In this Appendix, an outer-loop discrete-event-based controller for accommodating small randomly varying disturbances is designed. The designed outer-loop controller is layered on

top of the baseline controller, resulting in a better success rate of traversing random rough terrain in the detailed model simulations.

The remainder of this appendix is organized as follows: Section H.1 introduces the preliminary test results of the finite-state machine on randomly varying ground and discusses the main reason of failure. Section H.2 derives a linear time-invariant discrete-time model with exogenous input from the Poincaré map. Using the derived linear time-invariant discrete-time model, an outer-loop discrete event based control is designed, and discussed in Section H.3 with consideration of input saturation. The designed controller is tested in the detailed model to verify controller performance on random rough terrain. Section H.6 provides concluding remarks and future directions.

H.1 Performance of Finite-state Machine on Randomly Varying Ground

The performance of the finite-state machine on randomly varying ground is tested experimentally. Figure H.1 shows a snapshot taken from the experiment's video. In the experiment, MABEL was put in motion, walking on an initially flat floor. Boards with thickness of 1 inch (2.54 cm) and 2.5 inch (6.35 cm) were added randomly at every lap by two researchers. Four and half laps were completed before MABEL fell backward. The video is available at [75].

Figure H.2 shows MABEL's hip velocity. When MABEL stepped onto an obstacle at 332.7 sec (the last solid blue circle), the robot's speed decreased to 0.2 m/sec. This drastic decrease in speed is caused by an increase in potential energy, which decreases kinetic energy. Hence, MABEL's speed became too slow to allow MABEL to complete its step.

H.2 Problem Statement

As revealed in the first walking test on randomly varying ground, the main reason of failure is the controller's inability to regulate speed. Hence, an additional controller whose purpose is to attenuate the influence of disturbances in ground variations on the robot's speed is designed and will be discussed in the following sections. Before proceeding to a discussion of the design of additional controllers, we first refer to findings from biomechanical

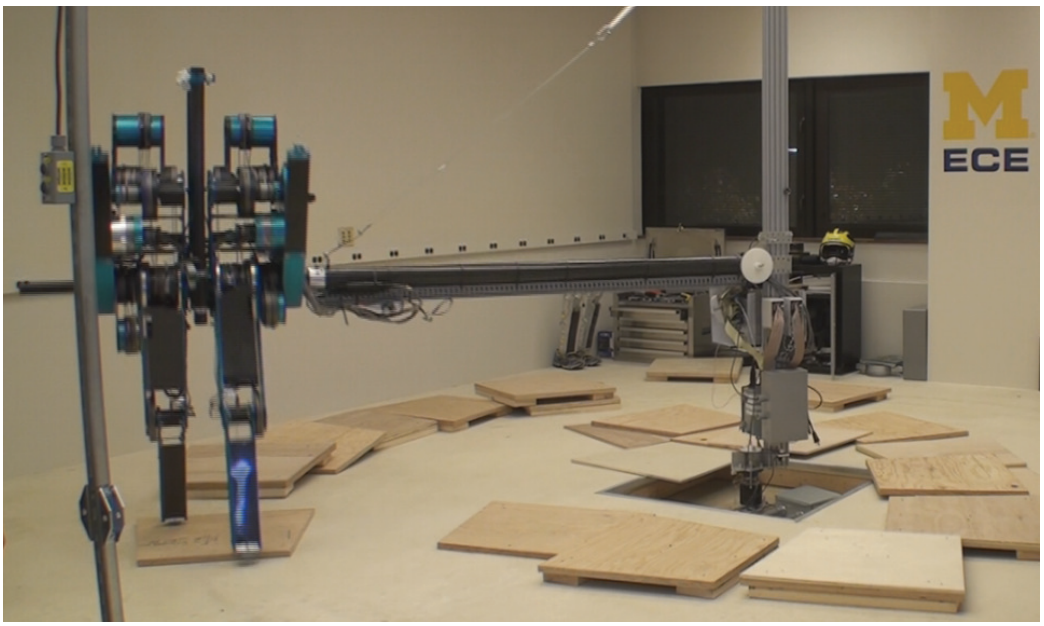


Figure H.1: MABEL is traversing randomly placed platforms.

studies.

A number of studies have shown that humans change gait patterns in response to velocity changes [102, 110]. For example, research studies in [102] have shown that humans change coordination patterns with respect to changes in walking speed, especially with motions of the trunk and head. The experiments performed in [110] also revealed that the relative phase as well as the range of trunk and pelvis rotation are changed as a function of walking speed. Drawing on these observations from human walking experiments, we will modify

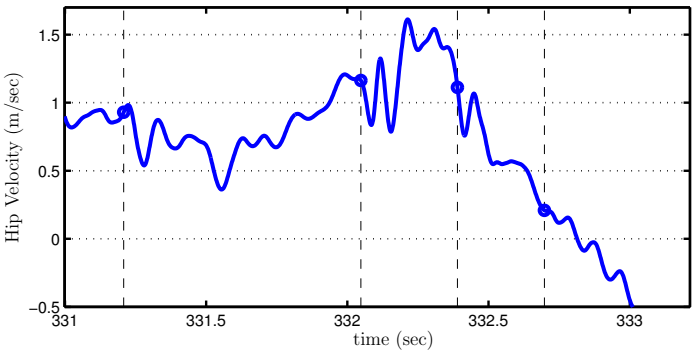


Figure H.2: Experimental data of randomly varying platform. Horizontal hip velocity is shown. The solid blue circles and black dashed lines display the end of the step. MABEL stepped upon the platform at 332.7sec (the last solid blue circle).

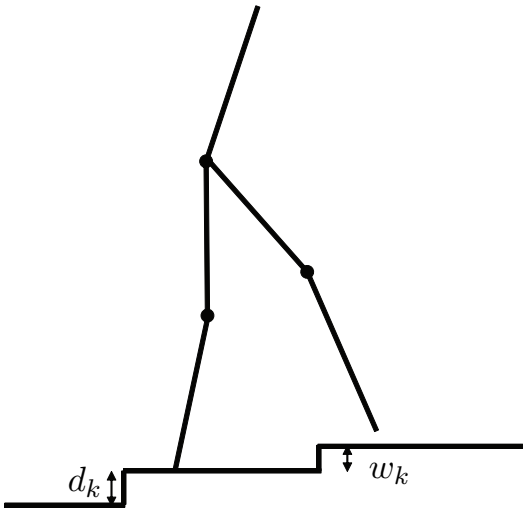


Figure H.3: Ground variation in height. w_k represents ground variation in the current step, and d_k represents ground variation from the previous step.

the trajectory of the torso, which is MABEL's counterpart of a human's trunk, as a means of changing and regulating gait speed in response to ground variation in height. These observations are aligned with MABEL's morphology because MABEL's torso represents approximately 65% of its weight, thereby heavily influencing MABEL's walking speed.

H.2.1 Ground Variation in Height: Step-to-step Disturbance

Ground variations in height affect the system dynamics only when the swing toe impacts with the ground. When the swing toe touches uneven ground with variations in height, MABEL's state after impact will vary from the fixed point value, and will affect the system throughout the step. Figure H.3 shows an illustration of a robot traversing uneven ground. $d_k := w_{k-1} \in \mathcal{W}$ is the previous step height and $w_k \in \mathcal{W}$ is the current step height, where $\mathcal{W} := [-\lambda, \lambda]$ and k represents step number. The ground variation from the previous step d_k changes the robot's post-impact state, which is denoted by x_k^+, thereby affecting the gait throughout the step, and also changing the robot's state at the end of the step. Along with the change in the robot's state at the end of the step, the current step's ground variation w_k will affect the ensuing step's post-impact state, which will be denoted by x_{k+1}^+.

H.2.2 Step-to-step Updates of Virtual Constraints

The virtual constraints for the regular-walking controller can be expressed by,

$$y = h(q_{\mathrm{s}}) = H_0 q_{\mathrm{s}} - h^d(s, \alpha^*) \tag{H.1}$$

where $H_0 q_{\mathrm{s}}$ are the controlled variables and given by,

$$H_0 q_{\mathrm{s}} = \begin{bmatrix} q_{\mathrm{LA_{sw}}} \\ q_{\mathrm{mLS_{sw}}} \\ q_{\mathrm{T}} \end{bmatrix}, \tag{H.2}$$

and $h^d(s, \alpha^*)$ are the desired trajectories of the virtual constraints and assembled as,

$$h^d\left(s, \alpha^*\right) = \begin{bmatrix} h^d_{\mathrm{LA_{sw}}}\left(s, \alpha\right) \\ h^d_{\mathrm{mLS_{sw}}}\left(s, \alpha\right) \\ h^d_{\mathrm{Tor}}\left(s, \alpha\right) \end{bmatrix}, \tag{H.3}$$

where α^* is the value of the Bézier coefficients corresponding to regular-walking.

In response to step-to-step disturbances, the virtual constraints are updated to attenuate the disturbances. These step-to-step updates on the virtual constraints are done by introducing control polynomials $h^{\mathrm{ctrl}}\left(s, \alpha_k^{\mathrm{ctrl}}\right)$ which are parametrized by Bézier coefficients α_k^{ctrl} that are updated step-to-step. This is obtained by modifying the virtual constraints at impact for each step k

$$y_k := h_k(q_{\mathrm{s}}) = H_0 q - h_*(s, \alpha^*) - h^{\mathrm{ctrl}}(s, \alpha^{\mathrm{ctrl}}\left(k\right)) \tag{H.4}$$

The output consists of the previous output (H.1), and an additional term h^{ctrl}. The additional term h^{ctrl} consists of a set of fifth-order Bézier polynomials which are assembled

as,

$$h^{\text{ctrl}}\left(s, \alpha^{\text{ctrl}}(k)\right) := \begin{bmatrix} h^{\text{ctrl}}_{\text{LA}_{\text{sw}}}(s) \\ h^{\text{ctrl}}_{\text{mLS}_{\text{sw}}}(s) \\ h^{\text{ctrl}}_{\text{Tor}}(s) \end{bmatrix}, \tag{H.5}$$

and parametrized by Bézier coefficients $\alpha^{\text{ctrl}}(k)$ which are in the form of,

$$\alpha^{\text{ctrl}}(k)\left(\alpha^1(k), \alpha^2(k)\right) := \alpha^1(k) \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} + \alpha^2(k) \begin{bmatrix} 0 & 0 & -0.3 & -0.7 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.3 & 0.7 & 1 & 1 \end{bmatrix}, \tag{H.6}$$

where $\alpha^1(k)$ and $\alpha^2(k)$ are the scalar variables which will be updated by the event-based controller at impact. This choice of Bézier coefficients leads to the virtual constraints of the torso and swing leg angle being modified in the opposite direction from each other by the same amount, keeping the virtual constraints on the absolute swing leg angle, which is simply the summation of torso angle and swing leg angle, unchanged, while the modification of the torso virtual constraint is made by updating $\alpha^1(k)$ and $\alpha^2(k)$.

As shown in Figure H.4(a), the scalar constant $\alpha^1(k)$ mainly modifies the torso virtual constraints at the beginning of the step. The reason for this modification is to facilitate the virtual constraints that effectively attenuate the rapid initial pitching of the torso due to the variations in ground height. As we have shown in the design of the step-down controller, the initial pitching forward of the torso due to the large step-down disturbance could be attenuated by setting the desired value further away from the pitching direction (see Figure 5.6 for step-down virtual constraints). We also have observed that MABEL's speed decreased when the robot steps onto the platform, as shown in the video of the first attempt at walking over rough ground [73]. This decrease in speed can be compensated by repositioning the center of mass of the robot. Because, for MABEL, the torso represents over 65% of the mass of the robot, we can easily move the position of the center of mass forward by leaning the torso forward at the beginning of the step. This change in torso

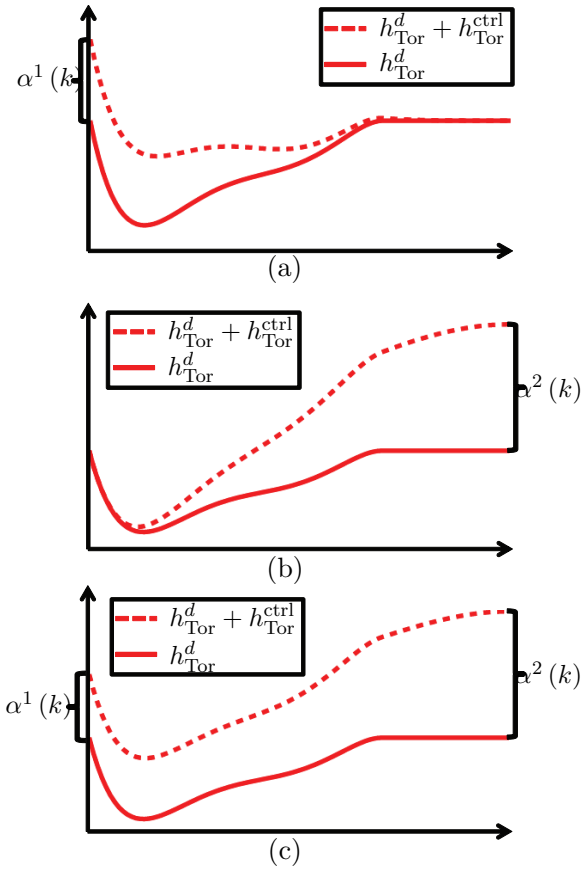


Figure H.4: Example of modified output. (a) $\alpha^1(k) \neq 0, \alpha^2(k) = 0$ (b) $\alpha^1(k) = 0, \alpha^2(k) \neq 0$ (c) $\alpha^1(k) \neq 0, \alpha^2(k) \neq 0$

angle can be made by updating $\alpha^1(k)$.

Furthermore, $\alpha^2(k)$ particularly changes the position of the torso at the end of the step (See Figure H.4(b)). Because the torso is heavy and the center of mass is not aligned with the hip joint, the pre-impact torso position heavily influences the speed of the ensuing step. When impact occurs with the torso being more forward than the nominal value, MABEL gains speed, and vice versa. Therefore, changing the pre-impact torso position provides essential control authority to regulate the speed.

The scalar constants $\alpha^1(k)$ and $\alpha^2(k)$ are assembled as $u(k) \in \mathcal{U}$,

$$u(k) := \begin{bmatrix} \alpha^1(k) \\ \alpha^2(k) \end{bmatrix}. \tag{H.7}$$

We also define $v(k) := u(k-1) \in \mathcal{U}$ which is merely the previous step's $u(k)$ for later use.

The step-to-step modification on the torso virtual constraint introduced in this section will be used as a control authority to attenuate step-to-step disturbances. In order to update u_k for each step, an outer-loop discrete event-based controller Γ^u is designed, as discussed in the following sections.

H.2.3 Poincaré Map and Its Linearization

An outer-loop discrete event-based control design was proposed to stabilize planar hopping in [82], 3D walking in [15], or running in [98]. While the main purpose of these event-based control designs is to stabilize the periodic orbit, an outer-loop discrete event-based controller will be designed here to attenuate the influence of ground height disturbance to the robot's speed with stability.

H.2.3.1 Construction of Discrete Nonlinear System

The full-order Poincaré map for the detailed model is considered for the design of the Γ^u robust outer-loop event-based controller. Unlike the design of the event-based controller performed in [82], [15], and [98], both step-to-step variations in the ground height of the current and previous step, which are $w(k)$ and $d(k)$ affects $x(k+1)$. Similar to the design of the speed controller performed in [117], previous control disturbance $v(k)$ as well as current control input $u(k)$ are considered for the construction of the Poincaré map. Therefore, the Poincaré map is defined as $P_{u(k),v(k),w(k),d(k)} : \mathcal{S}_{u(k),v(k),w(k),d(k)} \times \mathcal{U}^2 \times \mathcal{W}^2 \to \mathcal{S}_{u(k+1),v(k+1),w(k+1),d(k+1)}$, with $\mathcal{S}_{u(k),v(k),w(k),d(k)} := \mathcal{S}_{\mathrm{d}}^{w(k)}$, such that

$$x_{\mathrm{d}}^{+}(k+1) = P\left(x_{\mathrm{d}}^{+}(k), u(k), v(k), w(k), d(k)\right), \tag{H.8}$$

and thus, the following multi-input, single-output dynamic system with an exogenous input w_k can be defined as,

$$\begin{aligned} x_{\mathrm{d}}^{+}(k+1) &= P\left(x_{\mathrm{d}}^{+}(k), u(k), v(k), w(k), d(k)\right) \\ v(k+1) &= u(k) \end{aligned} \tag{H.9}$$

$$d(k+1) = w(k)$$
$$\eta(k) = h\left(x_{\mathrm{d}}^{+}(k), u(k), v(k), w(k), d(k)\right),$$

where output $\eta(k) \in \mathbb{R}$ is the value to be regulated, and given by,

$$\eta(k) := \dot{\theta}^{+} + \frac{1}{2}\dot{q}_{\mathrm{Tor}}^{+}. \tag{H.10}$$

The reason for this choice of $\eta(k)$ is as follows: hip velocity is an important value to be regulated so the robot does not fall back when stepping on the platform as shown in the experimental results in Section H.1. $\dot{\theta}^{+}$ roughly represents hip velocity at the beginning of the gait when multiplied by the leg length of 1 m, thereby becoming the first term of the $\eta(k)$. Furthermore, step-down experimental results in Section 5.1 show that large torso oscillation induced by the step-down event was the main reason for the failure of that experiment. Therefore, in order to regulate torso angular velocity at the beginning of the gait which can be the main source of large torso oscillation throughout the step when not properly regulated, torso angular velocity $\dot{q}_{\mathrm{Tor}}^{+}$ is added to $\dot{\theta}^{+}$ with a weight factor of $\frac{1}{2}$.

Let $\xi(k) := \left(x_{\mathrm{d}}^{+}(k) - x_{\mathrm{d}}^{+*}v(k), d(k)\right)$ and $y(k) := \eta(k) - \eta^{*}$, where x_{d}^{+*} and η^{*} correspond to the fixed point value. Then, rewriting the dynamic system in H.9 above with respect to $\xi(k)$ and $y(k)$ yields the following standard nonlinear discrete system with control input $u(k)$, exogenous input $w(k)$, and controlled output $y(k)$,

$$\xi(k+1) = \bar{P}(\xi(k), u(k), w(k))$$
$$y(k) = \bar{h}(\xi(k), u(k), w(k)) \tag{H.11}$$

Because the analytical expression for $\bar{P}$ does not exist in most cases, $\bar{P}$ needs to be approximated. In this dissertation, $\bar{P}$ will be approximated by a linear function of the states.

H.2.3.2 Linearization of Nonlinear Discrete System

Among a choice of many mathematical ways for linearizing a general nonlinear system, we use linear regression methods to linearize the system[1]. The linearization problem is to estimate the matrices of the system,

$$\begin{aligned} \xi(k+1) &= A\xi(k) + Bu(k) + B_w w(k) \\ y(k) &= C\xi(k) + Du(k) + Ew(k), \end{aligned} \tag{H.12}$$

given the data set of $\xi(k+1)$, $\xi(k)$, $u(k)$, $w(k)$, and $y(k)$. In order to obtain sufficiently rich data sets, we performed a significant number of separated two-step simulations with randomly perturbed states $\xi(k)$, input $u(k)$, and exogenous inputs $w(k)$. We then define fat matrices with n columns, where n represents the total number of simulations, and X_{k+1}, X_k, U_k, and W_k by horizontally concatenating each simulation's data $\xi^T(k+1)$, $\xi^T(k)$, $u^T(k)$ and $w^T(k)$, respectively. Because $y(k)$ is defined by the simple linear combination of $\xi(k)$, estimation of C, D, and E is a trivial problem. Then, the matrices A, B, and B_w are minimizers of the following quadratic cost,

$$J := \left\| X_{k+1} - \begin{bmatrix} A & B & B_w \end{bmatrix} \begin{bmatrix} X_k \\ U_k \\ W_k \end{bmatrix} \right\|^2, \tag{H.13}$$

and obtained by,

$$\begin{bmatrix} A & B & B_w \end{bmatrix} \approx X_{k+1} \begin{bmatrix} X_k \\ U_k \\ W_k \end{bmatrix}^+ \tag{H.14}$$

where V^+ denotes the Moore-Penrose pseudoinverse of V.

The controller will be designed on multi-input, single-output linear time-invariant system (H.12) with coefficient matrices obtained from (H.14).

[1]This method was suggested to us by Dr. Shai Revzen, University of Pennsylvania.

H.3 Controller Design

Controller design which attenuates the influence of the exogenous input w on the output y while achieving closed-loop stability for the system (H.12) will be studied in this section. The disturbance attenuation problem can be described as being one where the H^∞ norm of system (H.12) which is denoted as $\|T_{wy}\|_\infty$, where T_{wy} represents the transfer function from the disturbance w to the output y, satisfies the following inequality,

$$\|T_{wy}\|_\infty \leq \gamma \tag{H.15}$$

for some prescribed disturbance attenuation level γ. For a linear time-invariant system, the inequality (H.15) holds if and only if the following inequality holds [58],

$$\frac{\|y\|_{\mathcal{L}_2}}{\|w\|_{\mathcal{L}_2}} \leq \gamma \qquad \text{for } \|w\|_{\mathcal{L}_2} \neq 0. \tag{H.16}$$

The inequality (H.16) will be used to verify simulation results.

Control input u which allows inequality (H.15) is sought here, and we seek a controller in the form of linear state feedback $u(k) = K\xi(k)$, where K is the gain matrix. The closed-loop system with linear state feedback becomes,

$$\begin{aligned} \xi(k+1) &= A\xi(k) + BK\xi(k) + B_w w(k) \\ y(k) &= C\xi(k) + Du(k) + Ew(k), \end{aligned} \tag{H.17}$$

In consideration of the reasons set forth below, control input $u(k)$ will be saturated with some saturation level ν.

(a) The linearized system (H.17) is valid for only a small domain around the fixed point. A Large value of control input $u(k)$ will degrade the accuracy of the linearized model to predict the original system (H.11).

(b) A large $u(k)$ causes abrupt changes of the original virtual constraints. These abrupt changes easily cause large tracking errors, and actuator saturation.

With a saturated $u(k)$, the system (H.17) becomes,

$$\begin{aligned} \xi(k+1) &= A\xi(k) + Bsat_{\nu}(K\xi_k(k)) + B_w w(k) \\ y(k) &= C\xi(k), \end{aligned} \tag{H.18}$$

We employed techniques of quasilinear control (QLC) theory that have been developed through a number of research studies in the literature [34, 35, 26, 55] and summarized in [17] to design a controller on the system (H.18). The stochastically linearized version of the system (H.17) is,

$$\begin{aligned} \hat{\xi}(k+1) &= (A + BNK)\hat{\xi}_k + B_w w(k) \\ \hat{y}(k) &= C\hat{\xi}(k), \end{aligned} \tag{H.19}$$

where

$$N = \begin{bmatrix} \operatorname{erf}\left(\frac{\nu}{\sqrt{2}\sigma_{\hat{u}_1}}\right) & 0 \\ 0 & \operatorname{erf}\left(\frac{\nu}{\sqrt{2}\sigma_{\hat{u}_2}}\right) \end{bmatrix}, \tag{H.20}$$

and

$$\begin{bmatrix} \sigma^2_{\hat{u}_1} \\ \sigma^2_{\hat{u}_2} \end{bmatrix} = \|T_{wu}\|^2 \sigma_w^2 = KPK^T\sigma_w^2, \tag{H.21}$$

where P is the symmetric positive-definite matrix which satisfies,

$$(A + BNK)P(A + BNK)^T - P + B_w {B_w}^T = 0 \tag{H.22}$$

Using Theorem 4 of [21], we can easily show that $\|T_{wy}\|_{\infty}$ of system (H.19) is less than

γ if there exists symmetric positive definite matrix Q such that

$$\begin{bmatrix} Q & AQ+BL & B_w & 0 \\ (\cdot)^T & Q & 0 & Q^TC^T \\ (\cdot)^T & (\cdot)^T & I & 0 \\ (\cdot)^T & (\cdot)^T & (\cdot)^T & \gamma^2 I \end{bmatrix} > 0, \tag{H.23}$$

where $L = NKQ$. In addition, control gain K which satisfies (H.23) can be found using the following theorem.

Theorem H.1. *Consider system* (H.19) *subject to* (H.19)-(H.22)*. Then there exists K such that $\|T_{wy}\|_\infty$ is less than γ if there exists $L \in \mathbb{R}^{2\times n_\xi}$ and positive symmetric matrix $\bar{P}$ such that,*

$$\begin{bmatrix} \bar{P} & A\bar{P}+BL & B_w & 0 \\ (\cdot)^T & \bar{P} & 0 & \bar{P}^TC^T \\ (\cdot)^T & (\cdot)^T & I & 0 \\ (\cdot)^T & (\cdot)^T & (\cdot)^T & \gamma^2 I \end{bmatrix} > 0, \tag{H.24}$$

$$\begin{bmatrix} \frac{2\nu^2}{\pi\sigma_w^2} & L \\ (\cdot)^T & \bar{P} \end{bmatrix} > 0, \tag{H.25}$$

$$\begin{bmatrix} \bar{P} & A\bar{P}+BL & B_w \\ (\cdot)^T & \bar{P} & 0 \\ (\cdot)^T & (\cdot)^T & I \end{bmatrix} > 0. \tag{H.26}$$

Proof. By Theorem 4.3. of [17], (H.20), (H.21), and (H.22) has a solution $N > 0$ if and only if equalities (H.25) and (H.26) holds. Furthermore, equality (H.24) provides $\|T_{wy}\|_\infty < \gamma$ by Theorem 4 of [21]. □

Using this theorem, we can minimize the disturbance attenuation level γ for a given

saturation level ν. We calculated the minimum achievable γ, $\gamma_{\min}$, for saturation level $\nu \in [0.114, 11.43]$ deg. Figure H.5 shows the obtained $\gamma_{\min}$ and corresponding ν. It is clearly seen that $\gamma_{\min}$ is a decreasing function of ν, so there is a trade-off between $\gamma_{\min}$ and ν. Furthermore, the lower bound on $\gamma_{\min}$ is obtained with 9 degrees of saturation level ν. Therefore, ν cannot be made lower than the lower bound value even though saturation level is raised more than 9 degress. It is also observed that $\gamma_{\min}$ rapidly decreases down to the value of 5.14 degress, but, below that value, the decreasing ratio of $\gamma_{\min}$ becomes constant. Therefore, we set the saturation level ν at the value of 5.14 degrees where the benefit provided by increasing saturation level is maximized while decent attenuation level $\gamma_{\min}$ of 3.3 can be achieved. Based on the value of saturation level ν and corresponding $\gamma_{\min}$, control gain is calculated using Theorem H.1.

Given the obtained control gain, we simulated the system (H.18) with random disturbances $w(k) \in \mathcal{W}$. Figure H.6a shows ratios between the $\mathcal{L}_2$ norm of y and w. We can observe that the closed-loop system has the ratio less than the desired attenuation level γ (black dashed line), thereby satisfying the inequality (H.16), while the open-loop system does not. Figure H.6b displays control efforts to attenuate the disturbance. Although the control inputs were occasionally saturated with the saturation level ν, the controller successfully achieved desired disturbance attenuation performance by satisfying the desired attenuation level as shown in Figure H.6a. For purposes of verification, we carried out 500 simulations with different sequences of disturbances, and the results are shown in Figure H.7. We can observe that the desired disturbance level is achieved for all 500 simulations.

H.4 Controller Verification on Detailed Model

The outer-loop discrete event-based control design is now evaluated in the detailed model. The performances of the controller with and without the outer-loop update are not compared using the disturbance attenuation ratio here. Instead, we directly compare how many times the robot completed or failed to traverse a large number of terrain profiles with or without a discrete event-based control design. 150 test runs are performed in the simulation using controllers with and without outer-loop controller. Each test-run is

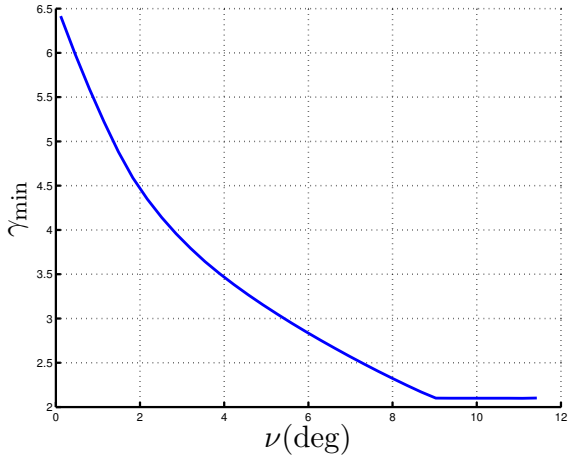


Figure H.5: Tradeoff locus of saturation level ν and $\gamma_{\min}$.

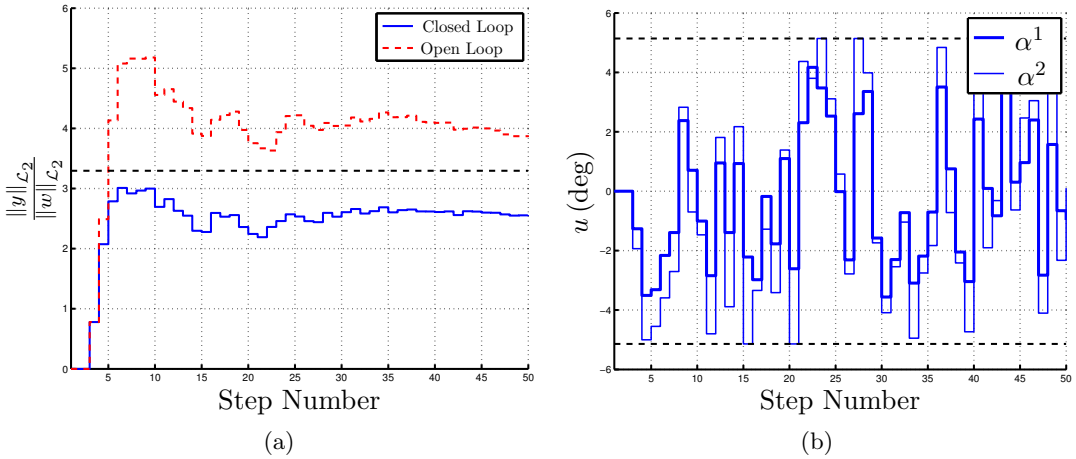


Figure H.6: Simulation results of the linearized system (H.18). (a) The ratio between $\mathcal{L}_2$ norm of w and y. (b) Control input u in deg.

conducted as follows: MABEL starts at a fixed point, walking on an initially flat floor. Rough terrain with ground variations in height are placed in front of MABEL. The terrain profiles consists of 12 steps with heights and depths from 1.5 cm to 5 cm and lengths from 35 cm to 65 cm as illustrated in Figure H.8. After 12 steps of random ground variations, the terrain becomes flat again. We then check whether the robot failed or completed to traverse given rough terrain, and, if it failed, the reason for the failure is further categorized into the following three groups: {*Falling Back, Tripping, Slipping*}.

Figure H.9 shows the simulation results. MABEL failed to complete 38% of test-runs without the outer-loop control design. On the other hand, the controller with the outer-

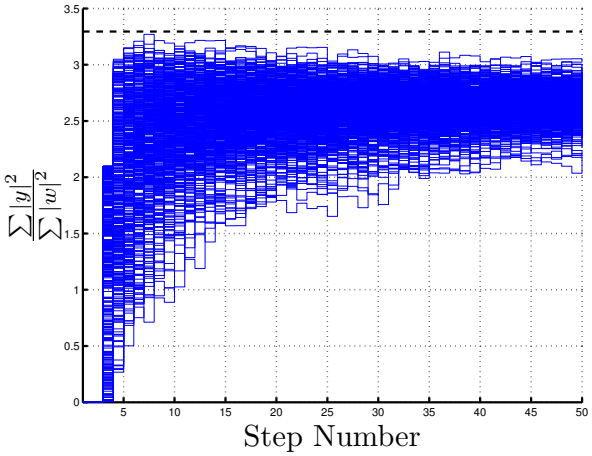


Figure H.7: The ratio between $\mathcal{L}_2$ norm of w and y of 500 simulation results.

loop control design was able to reduce the failure rate to 19% (see Figure H.9a). As shown in Figures H.9b, H.9c, and H.9d, the failure rate due to falling back is reduced to 6% by applying a discrete event-based control design, and the failure rate due to the tripping also decreased to 10%. However, more failures are caused by slipping. These failures due to slipping typically occur when the swing toe strikes the ground while the torso is leaning backward with a considerable velocity. Because the heavy torso rapidly leans backward, most of the robot's weight will be acting on the rear toe which is the stance toe, and normal component of the ground reaction force on the front toe, which is the swing toe striking the ground, will become very small at the moment of impact. Eventually, this small normal ground reaction force on the swing toe causes slippage between the front toe and the ground at the moment of impact. In particular, this slippage between the swing toe and the ground occurs when step-down disturbance is followed by a step-up event: the outer loop controller makes the torso lean backward to decrease the robot's speed, and early impact then occurs by encountering the step-up disturbance while the torso is leaning backward rapidly.

H.5 Addition of a Tripping Reflex Controller

An outer-loop controller focusing on attenuation of influence of disturbance has been designed and added onto the baseline controller. This section discusses the addition of a rapid-elevation strategy of the swing leg of the tripping reflex controller introduced in Chapter VI to further decrease the failure rate. We expect that the failure rate due to

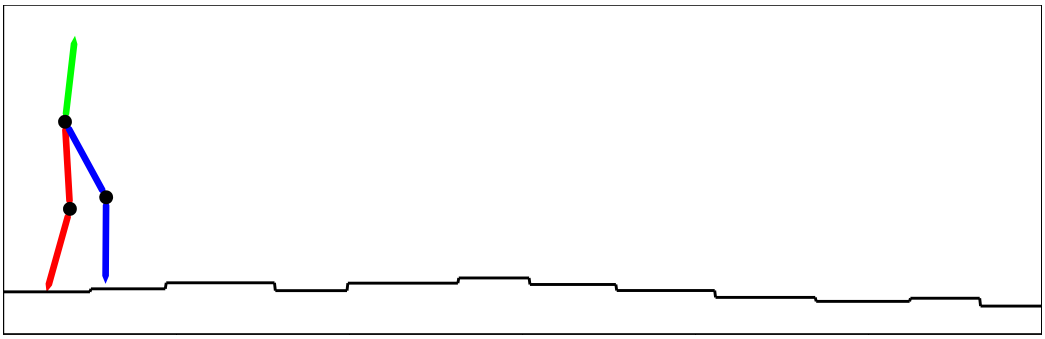


Figure H.8: Example terrain of test-run.

tripping significantly decreases by addition of rapid-elevation strategy of the swing leg. Using the same ground profiles as were used in the previous section, 150 simulations are conducted in the detailed model with the controller equipped with rapid-elevation of the swing leg . Figure H.10 illustrates the simulation results. The failure rate further decreased to 13% as shown in Figure H.10a. Remarkably, no failures are caused by tripping (see Figure H.10b), which shows excellent performance of the swing leg elevation strategy to reflex tripping. However, failures due to falling back increased by 3% compared to the Baseline+Outer loop case. Two thirds of this increase in failures due to falling back occurred when the robot walked over the terrain profiles which caused the tripping in Baseline+Outer loop case simulation. A closer look at the simulation data revealed that in those cases, the robot fell back after successfully stepping onto the obstacles by the use of the rapid-elevation strategy of the swing leg. To remedy these failures, we can try to increase saturation level ν, especially after swing leg rapid elevation, allowing the torso to learn further forward, or decrease the step width to allow the robot to gain speed after stepping onto the platforms.

H.6 Conclusion

This chapter has presented an outer-loop discrete event based control design for attenuation of disturbance effect on the output. A discrete time linear time-invariant system is obtained by linearizing the Poincaré map with respect to the fixed point. Techniques of quasilinear control are employed to design a controller which attenuates the effects of disturbances on the output with the desired level on the linearized model. Application of the outer-loop control design has resulted in a lower failure rate of traversing, given 150

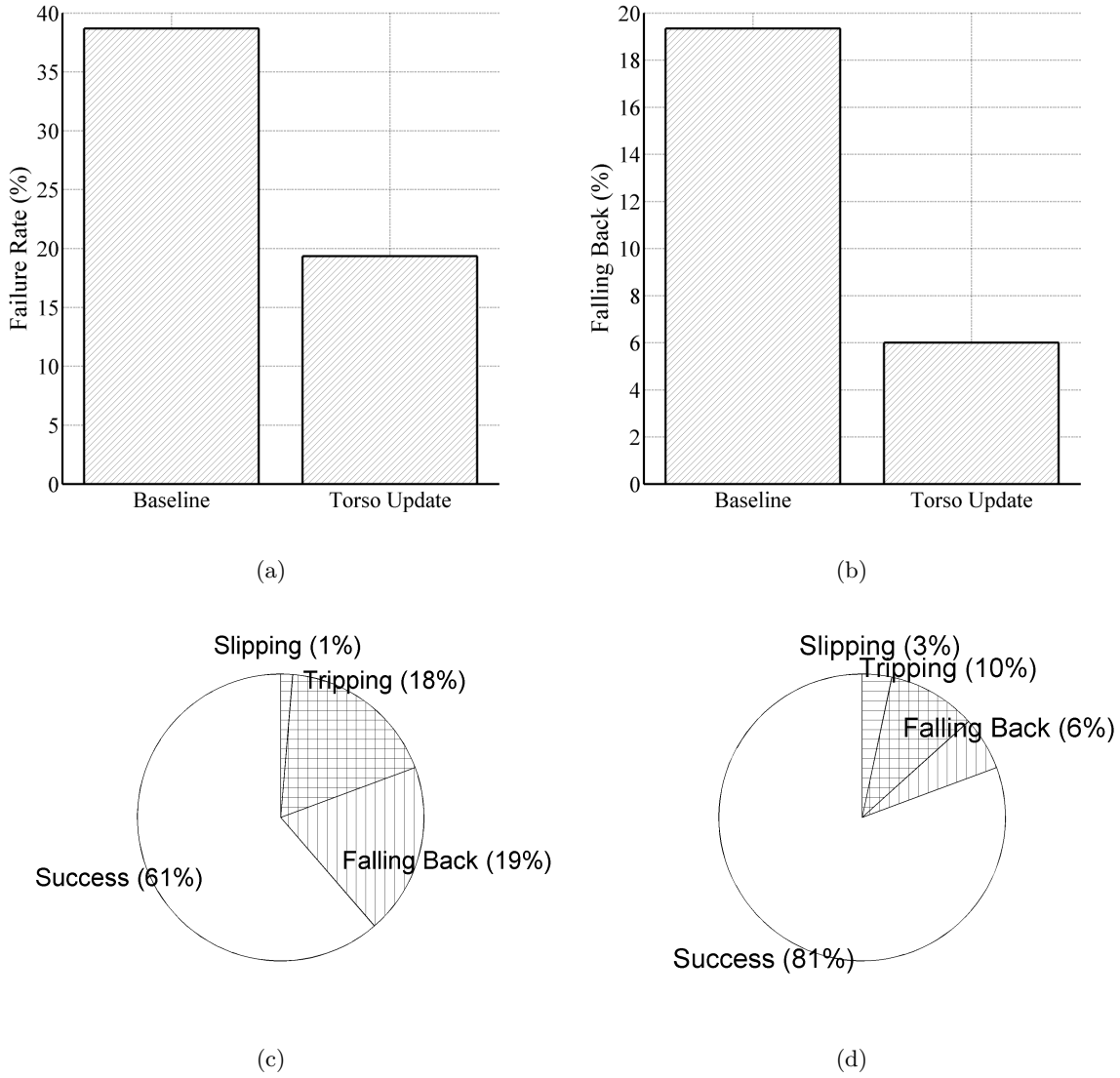


Figure H.9: Simulation data from the detailed model. (a) Total failure rate (b) Failure rate due to falling back (c) Pie chart of baseline controller (d) Pie chart of baseline + outer loop controller

terrain platforms. The failure rate was able to be further reduced by adding the rapid swing leg elevation strategy developed in Chapter VI for a reflex of tripping.

Though only 6% of total test-runs failed due to falling back in the simulation of the baseline + outer loop controller, research conducted in [87, 85, 120] provides hints that another decrease in failure rate due to falling back is possible. The idea of a capture point [87, 85] or foot placement estimator [120] provides a means to stop and stabilize a bipedal robot at a standing balance pose on one foot or two feet in response to an external

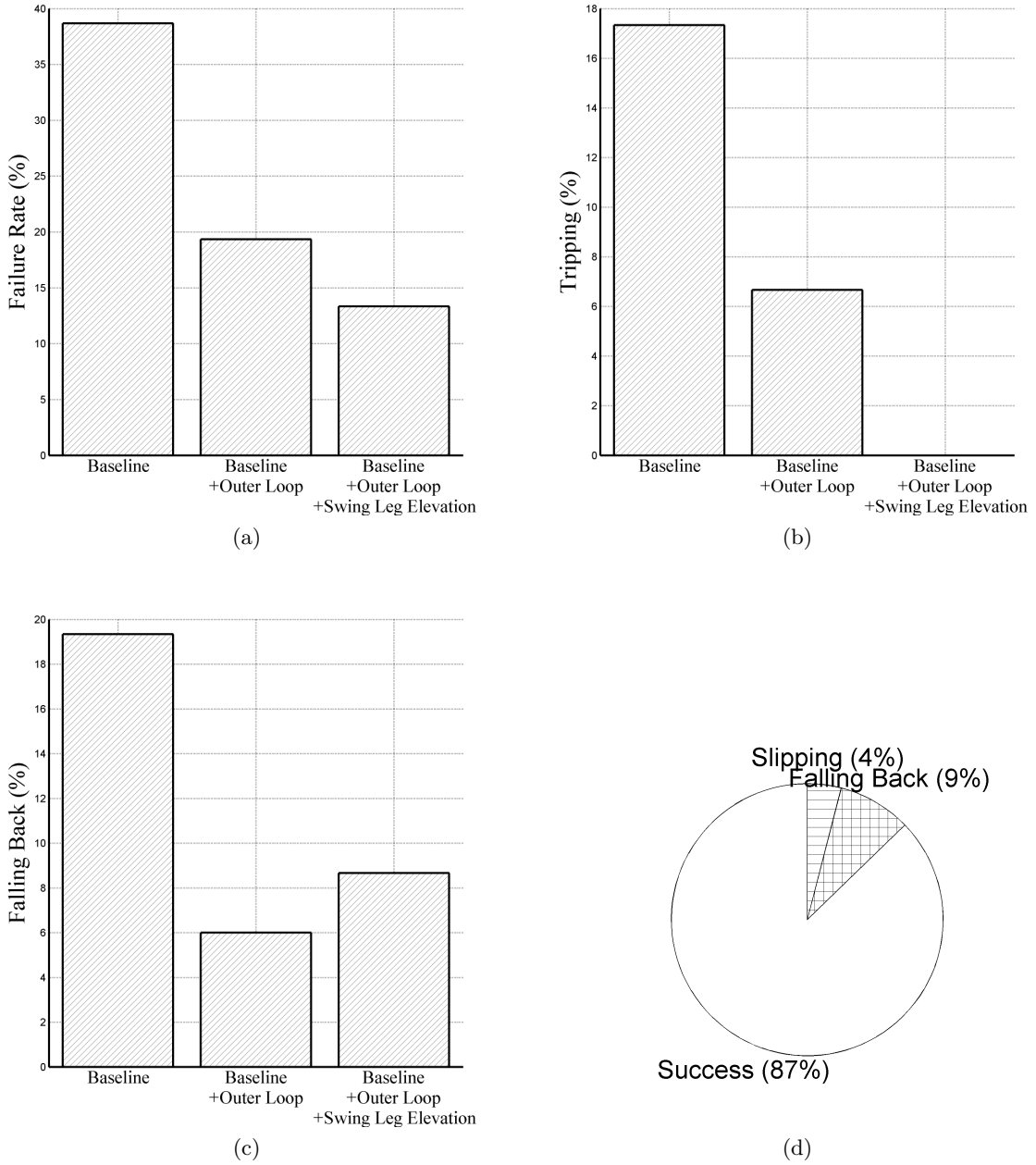


Figure H.10: Simulation data including swing leg elevation strategy from the detailed model. (a) Total failure rate (b) Failure rate due to tripping (c) Failure rate due to the falling back (d) Pie chart of baseline + outer loop controller + swing leg elevation

perturbation or sudden velocity change. These ideas can be also used to maintain the robot in a static pose when falling backward. From the static pose, gait can be re-initiated and continued.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] uBLAS - Basic Linear Algebra Subprograms library. [Online]. Available: http://www.boost.org

[2] The W prize. [Online]. Available: http://www.wprize.org

[3] A. Albert, M. Suppa, and W. Gerth, "Detection of stair dimensions for the path planning of a bipedal robot," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, vol. 2, 2001, pp. 1291 –1296 vol.2.

[4] R. M. Alexander, *Principles of Animal Locomotion*. Princeton University Press, 2003.

[5] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *The International Journal of Robotics Research*, vol. 5, no. 3, pp. 101–119, 1986.

[6] C. G. Atkeson and J. Morimoto, *Nonparametric Representation of Policies and Value Functions: A Trajectory-Based Approach*. MIT Press, 2003, p. 1643?1650.

[7] C. Baldon, I. Melchior, and J. Shulman, *Steps & stairways*. Rizzoli, 1989.

[8] G. N. Boone and J. K. Hodgins, "Slipping and tripping reflexes for bipedal robots," *Autonomous Robots*, vol. 4, pp. 259–271, July 1997.

[9] R. R. Burridge, A. A. Rizzi, and D. E. Koditschek, "Sequential composition of dynamically dexterous robot behaviors," *The International Journal of Robotics Research*, vol. 18, no. 6, pp. 534–555, 1999.

[10] K. Byl and R. Tedrake, "Approximate optimal control of the compass gait on rough terrain," in *IEEE International Conference on Robotics and Automation*, may 2008, pp. 1258 –1263.

[11] ——, "Dynamically diverse legged locomotion for rough terrain," in *IEEE International Conference on Robotics and Automation*, may 2009, pp. 1607 –1608.

[12] ——, "Metastable walking machines," *International Journal of Robotics Research*, vol. 28, pp. 1040–1064, August 2009.

[13] C. Canudas de Wit, H. Olsson, K. Astrom, and P. Lischinsky, "A new model for control of systems with friction," *IEEE Transactions on Automatic Control*, vol. 40, no. 3, pp. 419–425, Mar. 1995.

[14] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. Westervelt, C. Canudas-De-Wit, and J. Grizzle, "RABBIT: a testbed for advanced control theory," *IEEE Control Systems Magazine*, vol. 23, no. 5, pp. 57–79, Oct. 2003.

[15] C. Chevallereau, J. Grizzle, and C.-L. Shih, "Asymptotically stable walking of a five-link underactuated 3-d bipedal robot," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 37 –50, feb. 2009.

[16] C.-M. Chew, J. Pratt, and G. Pratt, "Blind walking of a planar bipedal robot on sloped terrain," in *IEEE International Conference on Robotics and Automation*, vol. 1, 1999, pp. 381 –386 vol.1.

[17] S. Ching, Y. Eun, C. Gokcek, P. Kabamba, and S. Meerkov, *Quasilinear Control: Performance Analysis and Design of Feedback Systems with Nonlinear Sensors and Actuators*. Cambridge University Press, 2010.

[18] A. Dabroom and H. Khalil, "Numerical differentiation using high-gain observers," in *IEEE Conference on Decision and Control*, San Diego, CA, USA, 1997, pp. 4790–4795.

[19] M. A. Daley and A. A. Biewener, "Running over rough terrain reveals limb control for intrinsic stability," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 42, pp. 15 681–15 686, 2006.

[20] M. A. Daley, J. R. Usherwood, G. Felix, and A. A. Biewener, "Running over rough terrain: guinea fowl maintain dynamic stability despite a large unexpected change in substrate height," *Journal of Experimental Biology*, vol. 209, no. 1, pp. 171–187, 2006.

[21] M. de Oliveira, J. Geromel, and J. Bernussou, "An LMI optimization approach to multiobjective controller design for discrete-time systems," in *IEEE Conference on Decision and Control*, vol. 4, 1999, pp. 3611 –3616 vol.4.

[22] C. C. de Wit, "On the concept of virtual constraints as a tool for walking robot control and balancing," *Annual Reviews in Control*, vol. 28, no. 2, pp. 157–166, 2004.

[23] S. Diop, J. W. Grizzle, P. E. Moraal, and A. Stefanopoulou, "Interpolation and numerical differentiation for observer design," in *American Control Conference*, Baltimore, MD, USA, 1994, pp. 1329–1333.

[24] J. J. Eng, D. A. Winter, and A. E. Patla, "Strategies for recovery from a trip in early and late swing during human walking," *Experimental Brain Research*, vol. 102, pp. 339–349, 1994.

[25] ——, "Intralimb dynamics simplify reactive control strategies during locomotion," *Journal of Biomechanics*, vol. 30, no. 6, pp. 581 – 588, 1997.

[26] Y. Eun, P. Kabamba, and S. Meerkov, "Analysis of random reference tracking in systems with saturating actuators," *IEEE Transactions on Automatic Control*, vol. 50, no. 11, pp. 1861 – 1866, nov. 2005.

[27] J. M. Fitch, "The dimensions of stairs," *Scientific American*, October 1974.

[28] Y. Fujimoto and A. Kawamura, "Robust biped walking with active interaction control between robot and environment," in *4th International Workshop on Advanced Motion Control*, vol. 1, mar 1996, pp. 247 –252 vol.1.

[29] E. Garcia and P. G. De Santos, "An improved energy stability margin for walking machines subject to dynamic effects," *Robotica*, vol. 23, no. 1, pp. 13–20, 2005.

[30] E. Garcia, J. Estremera, and P. G. De Santos, "A comparative study of stability margins for walking machines," *Robotica*, vol. 20, no. 06, pp. 595–606, 2002.

[31] E. J. Garcia Armada, Elena and P. Gonzalez de Santos, "A classification of stability margins for walking robots," in *International conference on Climbing and Walking Robots*, 2002.

[32] M. Gautier and W. Khalil, "On the identification of the inertial parameters of robots," in *IEEE Conference on Decision and Control*, Tampa, FL, USA, Dec. 1988, pp. 2264–2269.

[33] H. Geyer and H. Herr, "A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 3, pp. 263 –273, june 2010.

[34] C. Gokcek, P. Kabamba, and S. Meerkov, "Disturbance rejection in control systems with saturating actuators," in *American Control Conference*, vol. 2, 2000, pp. 740 –744 vol.2.

[35] ——, "An LQR/LQG theory for systems with saturating actuators," *IEEE Transactions on Automatic Control*, vol. 46, no. 10, pp. 1529 –1542, oct 2001.

[36] J. Gray, *How animals move*, ser. Royal Institution Christmas lectures. University Press, 1953.

[37] ——, *Animal locomotion*, ser. World naturalist. Weidenfeld & Nicolson, 1968.

[38] R. D. Gregg and M. W. Spong, "Reduction-based control of three-dimensional bipedal walking robots," *The International Journal of Robotics Research*, vol. 29, no. 6, pp. 680–702, May 2010.

[39] S. Grimmer, M. Ernst, M. Günther, and R. Blickhan, "Running on uneven ground: leg adjustment to vertical steps and self-stability," *Journal of Experimental Biology*, vol. 211, no. 18, pp. 2989–3000, 2008.

[40] J. W. Grizzle, "Publications on robotics." [Online]. Available: http://www.eecs.umich.edu/~grizzle/papers/robotics.html

[41] J. Grizzle, J. Hurst, B. Morris, H.-W. Park, and K. Sreenath, "MABEL, a new robotic bipedal walker and runner," in *American Control Conference*, St. Louis, MO, USA, June 2009, pp. 2030–2036.

[42] Y. Guan, E. S. Neo, K. Yokoi, and K. Tanie, "Stepping over obstacles with humanoid robots," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 958 –973, oct. 2006.

[43] K. Hashimoto, Y. Sugahara, H. Sunazuka, C. Tanaka, A. Ohta, M. Kawase, H. Lim, and A. Takanishi, "Biped landing pattern modification method with nonlinear compliance control," in *IEEE International Conference on Robotics and Automation*, Orlando, FL, USA, May 2006, pp. 1213–1218.

[44] S. Hirose, H. Tsukagoshi, and K. Yoneda, "Normalized energy stability margin and its contour of walking vehicles on rough terrain," in *IEEE International Conference on Robotics and Automation*, vol. 1, 2001, pp. 181 – 186 vol.1.

[45] K. Hitomi, T. Shibata, Y. Nakamura, and S. Ishii, "On-line learning of a feedback controller for quasi-passive-dynamic walking by a stochastic policy gradient method," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, aug. 2005, pp. 3803 – 3808.

[46] D. Hobbelen and M. Wisse, "A disturbance rejection measure for limit cycle walkers: The gait sensitivity norm," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1213 –1224, dec. 2007.

[47] ——, "Ankle actuation for limit cycle walkers," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 709–735, 2008.

[48] ——, "Swing-leg retraction for limit cycle walkers improves disturbance rejection," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 377 –389, april 2008.

[49] J. Hodgins and M. Raibert, "Adjusting step length for rough terrain locomotion," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 289–298, June 1991.

[50] Q. Huang and Y. Nakamura, "Sensory reflex control for humanoid walking," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 977 – 984, oct. 2005.

[51] Y. Hürmüzlü and T. Chang, "Rigid body collisions of a special class of planar kinematic chains," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 5, pp. 964–71, 1992.

[52] Y. Hurmuzlu and D. B. Marghitu, "Rigid Body Collisions of Planar Kinematic Chains With Multiple Contact Points," *The International Journal of Robotics Research*, vol. 13, no. 1, pp. 82–92, 1994.

[53] J. W. Hurst, "The role and implementation of compliance in legged locomotion," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, August 2008.

[54] J. Hurst, J. Chestnutt, and A. Rizzi, "Design and philosophy of the BiMASC, a highly dynamic biped," in *IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007, pp. 1863–1868.

[55] P. Kabamba, S. Meerkov, and C. Takahashi, "LMI approach to disturbance rejection in systems with saturating actuators," *University of Michigan Control Group Report*, no. CGR10-03, 2010.

[56] S. Kajita and K. Tani, "Adaptive gait control of a biped robot based on realtime sensing of the ground profile," in *IEEE International Conference on Robotics and Automation*, vol. 1, apr 1996, pp. 570 –577 vol.1.

[57] R. T. Katie Byl, "Metastable walking on stochastically rough terrain," in *Proceedings of Robotics: Science and Systems IV*, Zurich, Switzerland, June 2008.

[58] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.

[59] W. Khalil and E. Dombre, *Modeling, Identification and Control of Robots*. Bristol, PA, USA: Taylor & Francis, Inc., 2002.

[60] K. Kozlowski, *Modelling and Identification in Robotics*. Secaucus, NJ, USA: Springer-Verlag, 1998.

[61] C. Liu, C. G. Atkeson, and J. Su, "Biped walking control using a trajectory library," *Robotica*, 2010, submitted.

[62] L. Ljung, *System Identification: Theory for the User*, 1st ed. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1986, pp. 71–73.

[63] I. R. Manchester, U. Mettin, F. Iida, and R. Tedrake, "Stable dynamic walking over uneven terrain," *The International Journal of Robotics Research*, 2011.

[64] I. R. Manchester, M. M. Tobenkin, M. Levashov, and R. Tedrake, "Regions of attraction for hybrid limit cycles of walking robots." in *Proceedings of the 18th IFAC World Congress*, 2011.

[65] D. Marhefka and D. Orin, "Simulation of contact using a nonlinear damping model," in *IEEE International Conference on Robotics and Automation*, vol. 2, Minneapolis, MN, USA, April 1996, pp. 1662–1668.

[66] M. Mboup, C. Join, and M. Fliess, "A revised look at numerical differentiation with an application to nonlinear feedback control," *2007 Mediterranean Conference on Control Automation*, no. 1, pp. 1–6, 2007.

[67] T. Mcgeer, "Dynamic walking robots and the W prize [industry/research news]," *Robotics Automation Magazine, IEEE*, vol. 14, no. 2, pp. 13 –15, june 2007.

[68] R. Müller and R. Blickhan, "Running on uneven ground: Leg adjustments to altered ground level," *Human Movement Science*, vol. 29, no. 4, pp. 578 – 589, 2010.

[69] P. Michel, J. Chestnut, S. Kagami, K. Nishiwaki, J. Kuffner, and T. Kanade, "Gpu-accelerated real-time 3d tracking for humanoid locomotion and stair climbing," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 29 2007-nov. 2 2007, pp. 463 –469.

[70] J. Morimioto, G. Zeglin, and C. Atkeson, "Minimax differential dynamic programming: application to a biped walking robot," in *SICE 2003 Annual Conference*, vol. 3, aug. 2003, pp. 2310 –2315 Vol.3.

[71] J. Morimoto, J. Nakanishi, G. Endo, G. Cheng, C. Atkeson, and G. Zeglin, "Poincaré-map-based reinforcement learning for biped walking," in *IEEE International Conference on Robotics and Automation*, april 2005, pp. 2381 – 2386.

[72] M. Ogino, H. Toyama, and M. Asada, "Stabilizing biped walking on rough terrain based on the compliance control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, USA, Nov. 2007, pp. 4047–4052.

[73] H.-W. Park. (2010) First attempt at walking over rough ground for bipedal robot mabel. Youtube Video. [Online]. Available: http://youtu.be/IlWIWf4daNs

[74] ——. (2011) 8 inch (20 cm) blind step-down experiment on mabel. Youtube Video. [Online]. Available: http://youtu.be/fFlLlvOl7jo

[75] ——. (2011) Bipedal robot mabel walks over randomly varying ground: Experiment no. 1. Youtube Video. [Online]. Available: http://youtu.be/m8-j7s6-bzE

[76] ——. (2011) Mabel walking with feet. Youtube Video. [Online]. Available: http://youtu.be/cdsGz5GXOG0

[77] H.-W. Park, K. Sreenath, J. Hurst, and J. Grizzle, "Identification of a bipedal robot with a compliant drivetrain," *IEEE Control Systems Magazine*, vol. 31, no. 2, pp. 63 –88, april 2011.

[78] M. Pasupuleti, S. N. Yadukumar, and A. D. Ames, "Human-inspired underactuated bipedal robotic walking with amber on flat-ground, up-slope and uneven terrain," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, submitted.

[79] F. Pfeiffer and J. Holzl, "Parameter identification for industrial robots," in *IEEE International Conference on Robotics and Automation*, vol. 2, Nagoya, Japan, May 1995, pp. 1468–1476.

[80] F. Plestan, J. Grizzle, E. Westervelt, and G. Abba, "Stable walking of a 7-DOF biped robot," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 4, pp. 653–668, Aug. 2003.

[81] I. Poulakakis and J. Grizzle, "The spring loaded inverted pendulum as the hybrid zero dynamics of an asymmetric hopper," *IEEE Transactions on Automatic Control*, vol. 54, no. 8, pp. 1779 –1793, aug. 2009.

[82] I. Poulakakis and J. W. Grizzle, "Modeling and control of the monopedal robot thumper," in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 12-17 2009, pp. 3327–3334.

[83] M. J. Powell, H. Zhao, and A. D. Ames, "Motion primitives for human-inspired bipedal robotic locomotion: Walking and stair climbing," in *IEEE International Conference on Robotics and Automation*, dec. 2012, p. to appear.

[84] M. Powell, H. Zhao, and A. Ames, "Motion primitives for human-inspired bipedal robotic locomotion: Walking and stair climbing," in *IEEE International Conference on Robotics and Automation*, April 2012, to appear.

[85] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *IEEE-RAS International Conference on Humanoid Robots*, dec. 2006, pp. 200 –207.

[86] J. Pratt, C.-M. Chew, A. Torres, P. Dilworth, and G. Pratt, "Virtual model control: An intuitive approach for bipedal locomotion," *The International Journal of Robotics Research*, vol. 20, no. 2, pp. 129–143, 2001.

[87] J. E. Pratt and R. Tedrake, "Velocity-based stability margins for fast bipedal walking," *Fast Motions in Biomechanics and Robotics*, vol. 340, pp. 299–324, 2006.

[88] M. H. Raibert, *Legged Robots that Balance*. Cambridge, MA: MIT Press, 1986.

[89] C. Sabourin, O. Bruneau, and G. Buche, "Control strategy for the robust dynamic walk of a biped robot," *The International Journal of Robotics Research*, vol. 25, no. 9, pp. 843–860, Sept. 2006.

[90] A. W. Salatian, K. Y. Yi, and Y. F. Zheng, "Reinforcement learning for a biped robot to climb sloping surfaces," *Journal of Robotic Systems*, vol. 14, no. 4, pp. 283–296, 1997.

[91] U. Saranli, M. Buehler, and D. E. Koditschek, "RHex: A simple and highly mobile hexapod robot," *The International Journal of Robotics Research*, vol. 20, no. 1, pp. 616–631, July 2001.

[92] E. Schuitema, D. Hobbelen, P. Jonker, M. Wisse, and J. Karssen, "Using a controller based on reinforcement learning for a passive dynamic walking robot," in *IEEE-RAS International Conference on Humanoid Robots*, dec. 2005, pp. 232 –237.

[93] T. Shiratori, B. Coley, R. Cham, and J. K. Hodgins, "Simulating balance recovery responses to trips based on biomechanical principles," in *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '09. New York, NY, USA: ACM, 2009, pp. 37–46.

[94] A. Shkolnik, M. Levashov, I. R. Manchester, and R. Tedrake, "Bounding on rough terrain with the littledog robot," *The International Journal of Robotics Research*, 2010.

[95] R. W. Sinnet, M. J. Powell, R. P. Shah, and A. D. Ames, "A human-inspired hybrid control approach to bipedal robotic walking," in *18th IFAC World Congress*, 2011.

[96] G. Song and M. Zefran, "Stabilization of hybrid periodic orbits with application to bipedal walking," in *American Control Conference, 2006*, june 2006, p. 6 pp.

[97] M. W. Spong and F. Bullo, "Controlled symmetries and passive walking," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 1025–1031, 2005.

[98] K. Sreenath, "Feedback control of a bipedal walker and runner with compliance," Ph.D. dissertation, The University of Michigan, 2011.

[99] K. Sreenath, H.-W. Park, , and J. W. Grizzle, "Embedding active force control within the compliant hybrid zero dynamics to achieve stable, fast running on mabel," *The International Journal of Robotics Research*, 2012, submitted.

[100] K. Sreenath, H.-W. Park, I. Poulakakis, and J. W. Grizzle, "A compliant hybrid zero dynamics controller for stable, efficient and fast bipedal walking on MABEL," *The International Journal of Robotics Research*, vol. 30, no. 9, pp. 1170–1193, Aug 2011.

[101] M. Stilman, C. Atkeson, J. Kuffner, and G. Zeglin, "Dynamic programming in reduced dimensional spaces: Dynamic planning for robust biped locomotion," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, april 2005, pp. 2399 – 2404.

[102] V. Stokes, C. Andersson, and H. Forssberg, "Rotational and translational movement features of the pelvis and thorax during adult human locomotion," *Journal of Biomechanics*, vol. 22, no. 1, pp. 43 – 50, 1989.

[103] J. L.-S. Su and J. B. Dingwell, "Dynamic stability of passive dynamic walking on an irregular surface," *Journal of Biomechanical Engineering*, vol. 129, no. 6, pp. 802–810, 2007.

[104] J. Swevers, C. Ganseman, D. Tukel, J. de Schutter, and H. Van Brussel, "Optimal robot excitation and identification," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 5, pp. 730–740, Oct. 1997.

[105] J. Swevers, W. Verdonck, and J. D. Schutter, "Dynamic model identification for industrial robots," *IEEE Control Systems Magazine*, vol. 27, no. 5, pp. 58–71, Oct. 2007.

[106] T. Takum, S. Hayashi, and K. Hosoda, "3D bipedal robot with tunable leg compliance mechanism for multi-modal locomotion," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France, September 2008, pp. 1097–1102.

[107] W. Tan and A. Packard, "Stability region analysis using polynomial and composite polynomial lyapunov functions and sum-of-squares programming," *IEEE Transactions on Automatic Control*, vol. 53, no. 2, pp. 565 –571, march 2008.

[108] R. Tedrake, I. R. Manchester, M. Tobenkin, and J. W. Roberts, "LQR-trees: Feedback motion planning via sums-of-squares verification," *The International Journal of Robotics Research*, vol. 29, no. 8, pp. 1038–1052, 2010.

[109] T. Ueno, Y. Nakamura, T. Takuma, T. Shibata, K. Hosoda, and S. Ishii, "Fast and stable learning of quasi-passive dynamic walking by an unstable biped robot based on off-policy natural actor-critic," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, oct. 2006, pp. 5226 –5231.

[110] R. van Emmerik and R. Wagenaar, "Effects of walking velocity on relative phase dynamics in the trunk in human walking," *Journal of Biomechanics*, vol. 29, no. 9, pp. 1175 – 1184, 1996.

[111] P. Vandanjon, M. Gautier, and P. Desbats, "Identification of robots inertial parameters by means of spectrum analysis," in *IEEE International Conference on Robotics and Automation*, vol. 3, Nagoya, Japan, May 1995, pp. 3033–3038.

[112] Q. Wang, Y. Huang, J. Zhu, B. Chen, and L. Wang, *Dynamic Walking on Uneven Terrains with Passivity-Based Bipedal Robots*, ser. Lecture Notes in Electrical Engineering.

[113] S. Wang, J. Braaksma, R. Babuska, and D. Hobbelen, "Reinforcement learning control for biped robot walking on uneven surfaces," in *International Joint Conference on Neural Networks*, july 2006, pp. 4173 – 4178.

[114] E. R. Westervelt, "Toward a coherent framework for the control of planar biped locomotion," Ph.D. dissertation, University of Michigan, Jun. 2003.

[115] E. R. Westervelt, G. Buche, and J. W. Grizzle, "Experimental validation of a framework for the design of controllers that induce stable walking in planar bipeds," *The International Journal of Robotics Research*, vol. 24, no. 6, pp. 559–582, June 2004.

[116] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*. Taylor & Francis/CRC Press, 2007.

[117] E. R. Westervelt, G. Buche, and J. W. Grizzle, "Experimental validation of a framework for the design of controllers that induce stable walking in planar bipeds," *The International Journal of Robotics Research*, vol. 23, no. 6, pp. 559–582, June 2004.

[118] E. Whitman and C. Atkeson, "Control of a walking biped using a combination of simple policies," in *9th IEEE-RAS International Conference on Humanoid Robots*, dec. 2009, pp. 520 –527.

[119] P.-B. Wieber, "On the stability of walking systems," in *Proceedings of the International Workshop on Humanoid and Human Friendly Robotics*, Tsukuba, Japon, 2002.

[120] D. L. Wight, E. G. Kubica, and D. W. L. Wang, "Introduction of the foot placement estimator: A dynamic measure of balance for bipedal robotics," *Journal of Computational and Nonlinear Dynamics*, vol. 3, no. 1, p. 011009, 2008.

[121] M. Wisse, A. L. Schwab, R. Q. van der Linde, and F. C. T. van der Helm, "How to keep from falling forward: Elementary swing leg action for passive dynamic walkers," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 393–401, June 2005.

[122] J. Yamaguchi, N. Kinoshita, A. Takanishi, and I. Kato, "Development of a dynamic biped walking system for humanoid development of a biped walking robot adapting to the humans' living floor," in *IEEE International Conference on Robotics and Automation*, vol. 1, apr 1996, pp. 232 –239 vol.1.

[123] J. Yamaguchi, A. Takanishi, and I. Kato, "Experimental development of a foot mechanism with shock absorbing material for acquisition of landing surface position information and stabilization of dynamic biped walking," in *IEEE International Conference on Robotics and Automation*, vol. 3, may 1995, pp. 2892 –2899 vol.3.

[124] T. Yang, E. Westervelt, and A. Serrani, "A framework for the control of stable aperiodic walking in underactuated planar bipeds," in *IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007, pp. 4661–4666.

[125] Y. Zheng and J. Shen, "Gait synthesis for the sd-2 biped robot to climb sloping surface," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 1, pp. 86 –96, feb 1990.