

Feedback Control of a Bipedal Walker and Runner with Compliance

by

Koushil Sreenath

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2011

Doctoral Committee:

Professor Jessy W. Grizzle, Chair
Professor Anthony M. Bloch
Professor Arthur D. Kuo
Professor N. Harris McClamroch
Professor Semyon M. Meerkov

© Koushil Sreenath 2011

All Rights Reserved

To Sattvik, the l'il monster.

ACKNOWLEDGEMENTS

Well, I'm finally writing the acknowledgments. I would like to thank my advisor, Prof. Jessy Grizzle, for creating a wonderful opportunity for me to work on legged locomotion, for his dedication in nurturing me, for his patience and enthusiasm, for his fabulous intellectual support, and for giving me the freedom to explore ideas. I would like to thank my dissertation committee members, Prof. Anthony Bloch, Prof. Arthur Kuo, Prof. Harris McClamroch, and Prof. Semyon Meerkov, for their help and support. I would like to thank Hae-Won Park for his many roles as collaborator, co-author, travel companion, and friend throughout my years at the university. I would like to thank Ben Morris for his role as a fantastic mentor during my early years, for his many theoretical contributions that I routinely employ to make my work easier, and for his excellent advice - "Go big, or go home," which helped get the running experiments rolling. I would like to thank Ioannis Poulakakis for providing inspiration, for creating the framework of compliant hybrid zero dynamics that I happily borrowed, for solving many of my technical difficulties, and for providing great help as I looked for a job. I would like to thank Jonathan Hurst for creating MABEL, which enabled this dissertation, for helping me with great advice during my early years, for placing his confidence in me by extending a job offer. I hope I have helped meet some of his objectives for MABEL. I would like to thank Alireza Ramezani for his mechanical insight, and his crazy ideas with hot glue, which helped keep the switches on the feet of MABEL, enabling experiments with long runs. I would like to thank Jeff Koncsol for his selfless dedication to our project. His weekly visits to the laboratory and the generous sharing of his engineering experience contributed invaluable to the experiments reported here. I would like to thank Gabriel Buche for his many contributions to the design of the electronics, power supply and safety interlock systems. His prior experience with RABBIT was instrumental in us arriving at a much safer and more functional test facility. I would like to thank my family, and my close friends, for their innumerable sacrifices, which enabled this work. Finally, I would like to thank MABEL, for surviving all the crazy experiments I did on her.

Koushil Sreenath
Ann Arbor, August 2011

TABLE OF CONTENTS

| | |
|---|-----|
| DEDICATION | ii |
| ACKNOWLEDGEMENTS | iii |
| LIST OF FIGURES | vii |
| LIST OF TABLES | x |
| LIST OF APPENDICES | xi |
| ABSTRACT | xii |
| CHAPTER | |
| I. Introduction | 1 |
| 1.1 MABEL | 2 |
| 1.2 Contributions | 3 |
| 1.3 Organization of the Thesis | 5 |
| II. Literature Survey | 8 |
| 2.1 The Zero Moment Point Criterion | 11 |
| 2.2 Dynamic Running | 12 |
| 2.3 Passive Dynamics | 13 |
| 2.4 Energy Efficient Locomotion | 14 |
| 2.5 Simple Models for Walking and Running | 15 |
| 2.6 Reduction-based Control Design | 17 |
| 2.7 Hybrid Zero Dynamics | 18 |
| 2.8 Compliant Hybrid Zero Dynamics | 19 |
| 2.9 Summary | 20 |
| III. Control-Oriented Model of MABEL | 21 |
| 3.1 Description of MABEL | 21 |
| 3.2 MABEL Model | 23 |
| 3.2.1 MABEL's Unconstrained Dynamics | 23 |

| | | |
|------------|---|-----------|
| 3.2.2 | Dynamics of Stance | 25 |
| 3.2.3 | Stance to Stance Transition Map | 27 |
| 3.2.4 | Hybrid Model of Walking | 31 |
| 3.2.5 | Dynamics of Flight | 31 |
| 3.2.6 | Stance to Flight Transition Map | 33 |
| 3.2.7 | Flight to Stance Transition Map | 36 |
| 3.2.8 | Hybrid Model of Running | 39 |
| IV. | Control Design for Walking | 40 |
| 4.1 | Virtual Constraint Design for Walking | 41 |
| 4.1.1 | Deciding What to Control | 41 |
| 4.1.2 | Specification of the Constraints | 42 |
| 4.1.3 | Stance Motor Leg-shape Virtual Constraint | 44 |
| 4.1.4 | Torso Virtual Constraint | 46 |
| 4.1.5 | Swing Leg Virtual Constraints | 46 |
| 4.1.6 | Discussion | 47 |
| 4.2 | Zero Dynamics for Walking | 48 |
| 4.3 | Event Transitions | 49 |
| 4.4 | Gait Design Through Optimization | 51 |
| 4.5 | Fixed Point for Walking | 52 |
| 4.6 | Closed-loop Design and Stability Analysis | 56 |
| 4.6.1 | A PD + Feedforward Controller | 56 |
| 4.6.2 | Hybrid Invariance | 57 |
| 4.6.3 | Robustness Study of PD Controller | 60 |
| V. | Walking Experiments | 63 |
| 5.1 | Experiments | 63 |
| 5.1.1 | Exp. 1: Nominal Walking at a Fixed Speed | 64 |
| 5.1.2 | Exp. 2: Demonstration of Robustness to Perturbations . . | 70 |
| 5.1.3 | Exp. 3: Efficient Walking | 70 |
| 5.1.4 | Exp. 4: Compliant Zero Dynamics Controller | 72 |
| 5.1.5 | Exp. 5: Fast Walking | 74 |
| 5.2 | Discussion of the Experiments | 75 |
| 5.2.1 | Asymmetry | 75 |
| 5.2.2 | Impact Model | 76 |
| 5.2.3 | Exp. 3: Efficient Walking | 77 |
| 5.2.4 | Exp. 5: Fast Walking | 78 |
| 5.2.5 | Cable Stretch | 79 |
| 5.2.6 | Zeroing the Virtual Constraints | 80 |
| VI. | Control Design for Running: Embedding Active Force Control within the Compliant Hybrid Zero Dynamics | 81 |
| 6.1 | Motivation for Control Design | 82 |
| 6.2 | Overview of the Control Method | 83 |
| 6.3 | Virtual Constraint Design for Stance | 84 |

| | | |
|---------------------|--|------------|
| 6.3.1 | Deciding What to Control | 85 |
| 6.3.2 | Specification of the Constraints | 86 |
| 6.4 | Stance Zero Dynamics | 87 |
| 6.5 | Active Force Control - Virtual Compliance | 89 |
| 6.6 | Virtual Constraint Design for Flight | 91 |
| 6.7 | Flight Zero Dynamics | 92 |
| 6.8 | Event Transitions | 94 |
| 6.9 | Gait Design Through Optimization | 96 |
| 6.10 | Fixed Point for Running | 97 |
| 6.11 | Closed-loop Design and Stability Analysis | 101 |
| 6.11.1 | Exponentially Stabilizing Outer-loop Controller | 102 |
| 6.11.2 | Domain of Attraction Enlarging Outer-loop Nonlinear Controller | 103 |
| 6.12 | Modifications for Experimental Implementation | 107 |
| VII. | Running Experiments | 109 |
| 7.1 | Experiments | 110 |
| 7.1.1 | Running with Passive Feet | 110 |
| 7.1.2 | Running with Point Feet | 118 |
| VIII. | Concluding Remarks | 127 |
| 8.1 | Summary of New Contributions | 127 |
| 8.2 | Perspectives on Future Work | 129 |
| APPENDICES | | 131 |
| BIBLIOGRAPHY | | 147 |

LIST OF FIGURES

| <u>Figure</u> | | |
|---------------|---|----|
| 1.1 | MABEL, an experimental testbed for bipedal locomotion. | 3 |
| 2.1 | Classification of prior research into relevant categories. | 9 |
| 3.1 | MABEL’s powertrain comprising three cable differentials. | 22 |
| 3.2 | Stance commutative diagrams | 26 |
| 3.3 | Stance to Stance commutative diagrams | 29 |
| 3.4 | Flight commutative diagrams | 32 |
| 3.5 | Stance to Flight commutative diagrams | 35 |
| 3.6 | Flight to Stance commutative diagrams | 37 |
| 4.1 | The coordinate system used for the linkage. | 43 |
| 4.2 | The general shape of the stance phase virtual constraints. | 45 |
| 4.3 | The hybrid system for walking. | 50 |
| 4.4 | Virtual constraints and configuration variables for nominal fixed point. . . | 53 |
| 4.5 | Leg shape and B_{spring} variables for the nominal fixed point. | 54 |
| 4.6 | Actuator torques corresponding to the nominal fixed point. | 54 |
| 4.7 | Swing leg height and vertical center of mass for the nominal fixed point. . | 54 |
| 4.8 | Vertical component of the ground reaction force for the nominal walking fixed point. | 55 |
| 4.9 | Power plot of fixed point obtained by optimizing c_{mt} | 55 |

| | | |
|------|--|-----|
| 4.10 | Output plots of controller subject to various model perturbations. | 61 |
| 4.11 | Torque plots of controller subject to various model perturbations. | 62 |
| 5.1 | Experimental setup of the bipedal testbed MABEL. | 64 |
| 5.2 | Tracking for the swing-leg virtual constraints for various controllers. | 66 |
| 5.3 | Tracking for the stance-leg virtual constraints for various controllers. | 67 |
| 5.4 | Motor torques for various controllers. | 68 |
| 5.5 | Evolution of q_{Bsp} for left and right legs in stance for Exp. 1. | 69 |
| 5.6 | Stance and swing B_{spring} evolution for nominal walking experiment. | 69 |
| 5.7 | Step speeds for experiment of walking with external perturbations. | 71 |
| 5.8 | Power plot for Exp. 3. | 72 |
| 5.9 | Power plot for the hand-tuned virtual constraints experiment. | 73 |
| 5.10 | B_{spring} evolution for the compliant zero dynamics controller in Exp. 4. | 74 |
| 5.11 | Experimental B_{spring} and Torso compared with nominal fixed point. | 77 |
| 5.12 | Cable stretch for fast walking experiment, Exp. 5. | 79 |
| 6.1 | Feedback diagram illustrating the running controller structure. | 84 |
| 6.2 | Virtual constraints and configuration variables for nominal fixed point. | 98 |
| 6.3 | Leg shape and B_{spring} variables for the nominal fixed point. | 99 |
| 6.4 | Actuator torques corresponding to the nominal fixed point. | 99 |
| 6.5 | Swing leg height and vertical center of mass for the nominal fixed point. | 100 |
| 6.6 | Ground reaction force for the nominal running fixed point. | 100 |
| 6.7 | Simulation of 5° perturbation in impact value of leg shape. | 104 |
| 7.1 | Transitioning from walking to running for MABEL with feet. | 116 |
| 7.2 | Typical running step for MABEL with feet. | 117 |
| 7.3 | Evolution of the robot coordinates for experiment of running with feet. | 119 |

| | | |
|------|---|-----|
| 7.4 | Tracking for the stance phase virtual constraints. | 120 |
| 7.5 | Tracking for the flight phase virtual constraints. | 120 |
| 7.6 | Motor torques for the running experiment with feet. | 120 |
| 7.7 | Cable stretch in the leg shape direction for the stance leg. | 121 |
| 7.8 | Parameter plots for the outer-loop controllers employed in running. | 122 |
| 7.9 | Transitioning from walking to running for MABEL with point feet. | 123 |
| 7.10 | Typical running step for MABEL with point feet. | 124 |
| 7.11 | Speed at each step for the running experiment with point feet. | 125 |
| 7.12 | Parameter plots for the outer-loop controllers employed in running. | 126 |
| C.1 | The status of the bipedal testbed when the author began work. | 139 |
| C.2 | MABEL testbed assembled and ready for controllers. | 139 |
| D.1 | Effect of location of $si \rightarrow sd$ transition on c_{mt} | 144 |
| D.2 | Effect of torso offset on c_{mt} | 145 |
| D.3 | c_{mt} for various values of torso offset from a slow walking experiment. | 146 |

LIST OF TABLES

Table

| | | |
|-----|--|-----|
| 5.1 | Effect of Impact map scaling on walking speed. | 77 |
| 5.2 | Efficiency numbers for various bipedal robots. | 78 |
| 5.3 | Top walking speeds of bipedal robots | 79 |
| B.1 | The list of independent parameters to be determined by optimization. . . | 137 |

LIST OF APPENDICES

Appendix

| | | |
|----|---|-----|
| A. | Bézier Polynomials for Subphases | 132 |
| B. | Optimization Details for Walking | 135 |
| C. | Robot Construction and System ID | 138 |
| D. | Improving Energy Efficiency Further | 142 |

ABSTRACT

Feedback Control of a Bipedal Walker and Runner with Compliance

by

Koushil Sreenath

Chair: Jessy W. Grizzle

This dissertation contributes to the theoretical foundations of robotic bipedal locomotion and advances the experimental state of the art as well. On the theoretical side, a mathematical formalism for designing provably stable, walking and running gaits in bipedal robots with compliance is presented. A key contribution is a novel method of force control in robots with compliance. The theoretical work is validated experimentally on MABEL, a planar bipedal testbed that contains springs in its drivetrain for the purpose of enhancing both energy efficiency and agility of dynamic locomotion. While the potential energetic benefits of springs are well documented in the literature, feedback control designs that effectively realize this potential are lacking. The methods of virtual constraints and hybrid zero dynamics, originally developed for rigid robots with a single degree of underactuation, are extended and applied to MABEL, which has a novel compliant transmission and multiple degrees of underactuation. A time-invariant feedback controller is designed such that the closed-loop system respects the natural compliance of the open-loop system and realizes exponentially stable walking gaits. A second time-invariant feedback controller is designed such that the closed-loop system not only respects the natural compliance of the open-loop system, but also enables active force control within the compliant hybrid zero dynamics and results in exponentially stable running gaits.

Several experiments are presented that highlight different aspects of MABEL and the

feedback design method, ranging from basic elements such as stable walking, robustness under perturbations, energy efficient walking to a bipedal robot walking speed record of 1.5 m/s (3.4 mph), stable running with passive feet and with point feet. On MABEL, the full hybrid zero dynamics controller is implemented and was instrumental in achieving rapid walking and running, leading upto a kneed bipedal running speed record of 3.06 m/s (6.8 mph).

CHAPTER I

Introduction

A central challenge in legged locomotion for machines is that of coordinating a multi-joint electromechanical system so that it realizes walking and running gaits that are stable, agile, energy efficient, and fast. One part of the difficulty in meeting this challenge is the high degree of freedom and overall dynamical complexity of typical legged robots. A second difficulty arises from underactuation. One source of underactuation is the impracticality of actuating each degree of freedom in a legged robot, due to the weight of the actuators, power budget, expense, and other factors. Underactuation can also arise from the unilateral nature of ground contact forces. Specifically, because feet cannot pull against the ground, large moments at the ankle can cause foot rollover, which typically increases the underactuation of a mechanism.

Animals are able to move with great elegance and efficiency, having solved the problem of limb coordination through appropriate biomechanics, hierarchical neuronal control, and adaptation. An objective of the legged robotics community is to realize similar capabilities in legged machines. However, it must be noted that the material and components available to an engineer for creating a bipedal robot are quite different from those provided by biology, and consequently, we intend to imitate only the capabilities and not necessarily the solutions that are present in nature.

The presence of a compliant element in biological and man-made bipedal systems has been argued to be not merely an important characteristic of running, but rather a defining feature of running, without which running is very difficult, or even impossible to realize [15]. Indeed, the absence of a compliant element contributed to the inability of sustaining

a stable running gait on RABBIT [132]. To overcome this, MABEL a bipedal testbed at The University of Michigan, was developed with a novel powertrain that introduces series compliance.

This thesis work is directed toward developing a mathematical formalism for designing provably stable, walking and running gaits in bipedal robots with compliance, and demonstrating control solutions for walking and running on MABEL. In the first part of the thesis, the analytical work on compliant hybrid zero dynamics that was initiated in [92] for monopedal hoppers is shown to be relevant for bipedal walkers. The novel contributions are primarily on the experimental side, where excellent robustness, speed, and energy efficiency are demonstrated on MABEL. In the second part of the thesis, which addresses running, active force control is incorporated into the mathematical formalism of the compliant hybrid zero dynamics. The importance of the analytical contribution is demonstrated through an experimental implementation of the controller that achieves stable running with passive feet and with point feet. The running that is realized has a very natural and elegant appearance with a significant part of the gait spent in flight and with good ground clearance.

1.1 MABEL

MABEL is a bipedal testbed at the University of Michigan. The robot is planar, with a torso, two legs with revolute knees, and four actuators. MABEL was designed to be both a robust walker and a fast runner. A detailed description of the robot has been presented in [59, 60, 58], and the identification of its dynamic model is reported in [50].

MABEL was designed by Jonathan Hurst as part of his doctoral research [58]. The robot's drivetrain uses a set of differentials to create a *virtual prismatic leg* between the hip and the toe such that one actuator controls the angle of the virtual leg with respect to the torso, and another actuator controls the length of the virtual leg. Moreover, the drivetrain also introduces a *compliant element*, a unilateral spring present in the transmission, that acts along the virtual leg in series with the actuator controlling the leg length. With this design, it is possible to place all of the actuators in the torso, thereby making the legs relatively light and enabling rapid leg motion. More details on the design philosophy are



Figure 1.1: MABEL, an experimental testbed for bipedal locomotion. The robot is planar, with a boom providing stabilization in the sagittal plane. The robot’s drivetrain contains springs for enhanced power efficiency.

available in [49, 58].

The MABEL bipedal testbed is being used to continue a control design philosophy initiated with RABBIT [16]: Hypothesis-driven (i.e., theorem-proof) control design methods are developed for a class of robots, and their validity is evaluated on the testbeds.

1.2 Contributions

The key results of the thesis are summarized next.

- (a) *Walking Control Design*: A Hybrid Zero Dynamics (HZD)-based controller is designed for walking such that the natural compliant dynamics is preserved in the closed-loop system (robot plus controller). This ensures that the designed walking gait uses the compliance to do negative work at impact, instead of it being done by the actuators, thereby improving the energy efficiency of walking. Stability analysis using the method of Poincaré is then carried out to check stability of the closed-loop system. Prior to experimentally testing the controller, simulations with various model perturbations are performed to establish robustness of the designed controller. The controller is then experimentally validated on MABEL.

- (b) *Energy Efficient Walking*: Walking gaits are designed to optimize the energetic cost of mechanical transport [26, 27]. This results in a gait that is more than twice as efficient on the testbed than a gait that we had designed by hand and reported in [49]. The resulting cost of mechanical transport is approximately three times more efficient than RABBIT, and 12 times better than Honda’s ASIMO, even though MABEL does not have feet. This puts MABEL’s energy efficiency within a factor of two of T.U. Delft’s Denise and within a factor of three of the Cornell Biped, none of which can step over obstacles or run; it is also within a factor of two of the MIT Spring Flamingo which can easily step over obstacles but cannot run, and within a factor of three of humans, who can do all of the above.
- (c) *Fast Walking*: In preparation for running experiments, fast walking is attempted, where each step may be on the order of 300 to 350 ms. Very precise control is needed for accurately implementing the virtual constraints of an HZD controller with these gait times. All experimental implementations of the virtual constraints reported to date have relied on local PD controllers [132]. The zero dynamics controllers provide great tracking accuracy in theory, but are often criticized for being overly dependent on high model accuracy, and for being too complex to implement in real-time. Here we demonstrate, for the first time, an experimental implementation of a compliant HZD controller. The tracking accuracy attained is far better than the simple PD controllers used earlier. With a zero dynamics controller, a top sustained walking speed of 1.5 m/s (3.4 mph) is attained experimentally.
- (d) *Running Controller Design*: A running controller is designed based on virtual constraints by creating a compliant hybrid zero dynamics. By defining only three virtual constraints (on a system with four actuators) in the stance phase of running, an actuated HZD is created which enables using force control within the HZD. With active force control, a virtual compliant element is created to enable active tuning of the physical compliance. An optimization problem is then posed to find a periodic running gait. Two outer-loop event based controllers are designed to (i) exponentially stabilize, and (ii) increase the domain of attraction of the limit cycle representing the periodic run-

ning gait. The designed controller is validated on a complex model of the system that incorporates cable stretch as part of the model. The controller is finally experimentally deployed.

- (e) *Running with Passive Feet*: With a modification to the hardware, by installing passive feet, stable running is demonstrated on MABEL with an experiment consisting of 100 consecutive running steps at an average speed of 1.07 m/s, with a corresponding flight phase that is 30% of the gait and an estimated ground clearance of 2 inches (5 cm).
- (f) *Running with Point Feet*: Stable running with point feet is demonstrated on MABEL with an experiment consisting of 113 consecutive running steps at an average speed of 1.95 m/s, and a peak speed of 3.06 m/s. The estimated ground clearance is 3 – 4 inches (7.5 – 10 cm). At 2 m/s, the flight phase is 35% of the gait, and at 3 m/s, the flight phase is 39% of the gait.

1.3 Organization of the Thesis

To effectively utilize compliance when present in bipedal robots, this thesis develops feedback controllers based on compliant hybrid zero dynamics for achieving stable walking and running. The control designs are experimentally validated on MABEL. With this goal in mind, the remainder of the thesis is organized as follows.

Chapter II presents a brief survey of relevant literature and places in perspective work presented in this thesis.

Chapter III presents the general features of MABEL’s morphology, and points out key ideas behind the design of the biped. Fully comprehending the philosophy behind the design is imperative, as it provides a means of constructing simpler models of the biped, and also provides intuition into designing controllers that respect the natural dynamics of the system. The mathematical model for the planar biped is developed, based on the Lagrangian framework. The full, unconstrained, nine-degree-of-freedom (DOF) model is obtained. Then, by imposing various holonomic constraints for the stance and flight phases, the corresponding constrained dynamics are obtained. The transition maps between the stance and flight phases are developed to model the corresponding discrete transitions.

Finally the continuous-time dynamics and the transition maps are assembled to arrive at hybrid models for walking and running. These models are used for control design.

Chapter IV provides a systematic procedure, based on virtual constraints, to design a suite of walking gaits. The idea behind the choice and design of the virtual constraints is presented and motivated. Optimization cost criteria are presented to both optimize for electrical energy consumed, and positive mechanical work done. The design and stability analysis of two stabilizing controllers to realize the designed gaits is presented next. These controllers are a simple feedforward-plus-PD controller, developed with the idea of simplicity in experimental implementation, and the full hybrid-zero-dynamics controller, which creates hybrid invariance. Finally, the robustness of the simple feedforward-plus-PD control to various model perturbations is studied to analyze experimental worthiness of the controller.

Chapter V presents the experimental testbed used for validating the designed controllers. Several experiments are presented to demonstrate the validity and robustness of the designed controllers. This includes experiments illustrating the walking efficiency and fast walking speed obtained through the designed controllers. Finally a discussion of various aspects of the robot and the feedback controllers revealed by the experiments is presented.

Chapter VI presents a control design for achieving stable running. Virtual constraints are presented for the stance phase of running that result in a restricted dynamics that is actuated. Virtual compliance is introduced as a means of varying the effective compliance of the system in the stance phase. Virtual constraints for the flight phase of running are presented along with a fixed point representing a periodic running gait. The controller structure is presented, which includes a continuous-time nonlinear controller based on compliant hybrid zero dynamics, and two outer-loop event based controllers for exponentially stabilizing and increasing the domain of attraction of the fixed point representing the periodic running gait. Finally, additional controller parameter modifications are presented that will address the cable stretch present in the experimental testbed.

Chapter VII presents experimental results to demonstrate the validity and robustness of the designed running controller.

Chapter VIII provides concluding remarks and briefly summarizes research accomplished in this dissertation.

Appendix A presents a framework of using Bézier polynomials for virtual constraints with subphases. Further details are provided on how to choose the coefficients for various subphases for all the virtual constraints for the stance phase of walking. Appendix B presents a sequence of steps to be followed as part of the optimization process to find fixed points. Various constraints that need to be satisfied are also presented. Appendix C presents an overview of the behind the scenes work that was carried out in constructing the bipedal testbed, and briefly summarizes the system identification process carried out to identify the robot model.

CHAPTER II

Literature Survey

A very broad spectrum of approaches and ideas exist for achieving stable bipedal locomotion. The aim of this chapter is to categorize and present a relevant cross-section of these approaches that serve to motivate and place the work presented in this thesis in perspective. Towards accomplishing this, prior research is broadly classified into categories and the categories are grouped into layers based on their relevance to the work presented in this thesis, as clearly illustrated in Figure 2.1. Here, we briefly describe each of the categories and in subsequent sections present the literature review.

First, we review broad categories that are relevant to the field of legged locomotion, but do not specifically address the work being presented in this thesis. Three classes of research in bipedal locomotion are considered: (a) The zero moment point criterion, (b) Dynamic running, and (c) Passive dynamics. To achieve upright, stable bipedal locomotion on existing humanoid robots, without having to address the difficulties in formulating analytical controllers for nonlinear, multi-phase, and hybrid models, several researchers resort to studying static or quasi-static walking using control schemes based on regulating the zero moment point (ZMP). This essentially boils down to making online modifications to the gait so as to maintain the center of pressure of the ground reaction forces on the stance foot strictly within the support polygon, and results in slow, flat-footed quasi-static walking gaits. ZMP based control schemes have also been used to make robots run, but the resulting running gait is distinctly robot-like, with short strides, small flight times, and little ground clearance. Early dynamically stable running robots on the other hand employed the natural dynamics of the system through simple intuitive controllers to achieve life-like running

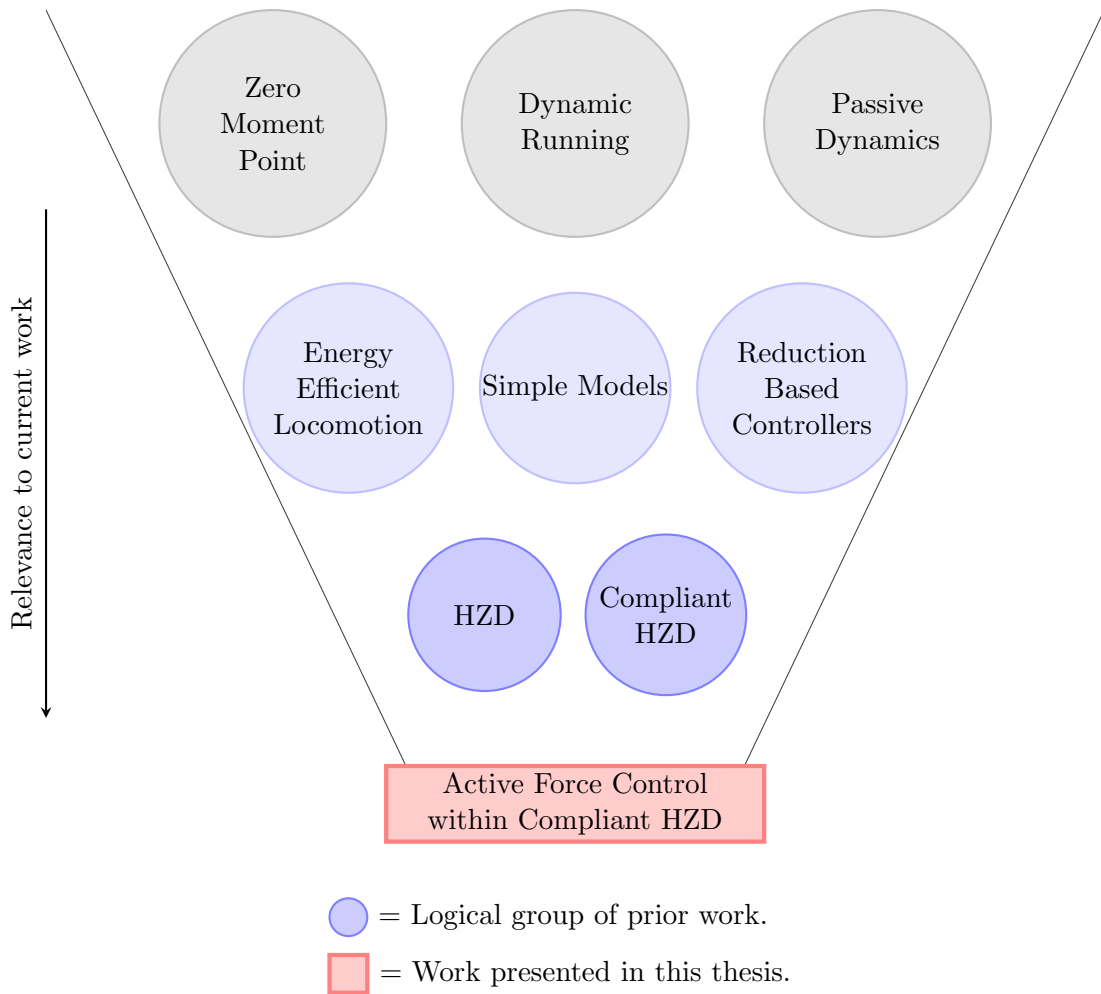


Figure 2.1: Prior research is broadly classified into categories and grouped into layers based on their relevance to the work presented in this thesis.

gaits. However, these machines usually have light legs with a trivial torso. Taking the idea of employing natural dynamics further, passive dynamics research deals with designing mechanical systems whose natural dynamics cause the system to evolve in such a manner so as to create dynamically stable walking or running gaits down gentle slopes. Although such machines are energy efficient, they are not robust to perturbations and are very sensitive to initial conditions.

Next, we review a narrower set of categories that relate more closely to a particular aspect of the work presented in this thesis: (a) Energy efficient locomotion, (b) Simple models for walking and running, and (c) Reduction-based control design. Based on the idea of passive dynamics for walking down gentle slopes, minimally actuated robots were developed that could walk on flat ground, have a larger domain of attraction in comparison to their passive counterparts, and still retain good energy efficiency. As another means of improving energy efficiency, compliance in the system has been introduced to minimize the loss of energy at impact. From biomechanical studies, animals are remarkably energy-efficient. To explain the remarkable similarity of ground reaction profiles measured in experiments with diverse animals, which suggests common energy-saving mechanisms in animals, researchers proposed simple underlying models, whose purpose is to capture the dominant locomotion behavior resulting from complex, high-dimensional, nonlinear, dynamically coupled interactions between an organism and its environment. These models have been extensively analyzed mathematically, and several controllers have been proposed to achieve stable walking and running gaits. On a similar note, with the promise of analytical tractability of simple models, researchers have tried to find properties, whose invariance (possibly through feedback) leads to the reduction of a higher-dimensional model to a lower-dimensional model, wherein the behavior of the higher-dimensional complex model can be explained by the behavior of the lower-dimensional simpler model and vice versa. This results in control design which is based on reduction.

Finally, we review literature on which the work presented in this thesis is directly based: (a) Hybrid zero dynamics, and (b) Compliant hybrid zero dynamics. Hybrid zero dynamics is a form of dynamic reduction through feedback control, where the feedback imposes virtual constraints and enables studying the stability properties of a higher-dimensional model by

studying the stability properties of a restricted (that is, respecting the virtual constraints) lower-dimensional model. An extension of hybrid zero dynamics on systems with compliance is to preserve the physical compliance of the system as part of the reduced system. This is achieved by creating a hybrid zero dynamics which is compliant.

The following sections examine each of the logical categories of work identified above in greater detail. Several other categories of research on achieving stable bipedal locomotion exist, such as neuronal control [76, 98], learning based controllers [102], sliding mode controllers [119], energy shaping controllers [113, 54] and port Hamiltonians based controllers [30]. These categories are not discussed further to keep the survey focused. For a more complete, exhaustive survey and history of dynamic robotic legged locomotion up until the present, see [118, 95, 126, 81, 56, 55, 130, 135, 17].

2.1 The Zero Moment Point Criterion

Early attempts at designing legged machines led to slow moving, statically stable robots. Static walking occurs when the ground projection of the center of mass of the system lies within the support polygon. This class of legged robots generally has large and heavy feet. Several robots of this type utilize the zero moment point¹ (ZMP) criterion to achieve a quasi-stable walking gait. The ZMP criterion is well surveyed in the anniversary paper [125], and states that when the ZMP is contained within the interior of the robot’s support polygon formed by the robot’s feet, the robot is stable, i.e., will not topple by rotating on an edge of its feet. Researchers have also used the ZMP as a simple metric for robustness. By looking at how close the ZMP is to the boundary of the support polygon, one can arrive at an estimate of how close the robot is to tipping over. Other related notions include the Foot Rotation Index (FRI) [41] and the center of pressure (CoP) [107].

The ZMP criterion has been employed to design walking gaits so as to regulate the ZMP to a desired value within the support polygon [65]. By keeping the ZMP within the support polygon, a degree of underactuation is avoided at the ground-foot interface. Some of the notable robots employing the ZMP criterion are Honda’s ASIMO [104], Sony’s QRIO [37],

¹The ZMP is defined to be the point on the ground where the net moment of the gravity and inertia forces on the robot is zero.

and the HRP series [68]. To be able to control the position of the ZMP, these robots need to be fully actuated.

With the ZMP-based control design, researchers do not have to address the difficulties in formulating analytical controllers for nonlinear, multi-phase, and hybrid models. However, the resulting walking gaits are flat-footed, generally not robust to external perturbations (a modest push could saturate the ankle actuators, resulting in an inability to regulate the position of the ZMP) and the achieved energy efficiency is low [27]. Furthermore, as shown by Choi [22], the satisfaction of the ZMP criterion is not a sufficient condition for stability. Neither is it necessary for stability as shown by analysis and walking experiments on RABBIT [132], and MABEL [115].

The ZMP criterion has also been employed to demonstrate running on Sony’s QRIO [86], Honda’s ASIMO, Toyota’s humanoid robot [116] (with running at a top speed of 1.94 m/s), HRP-2LR [67] and HRP-2LT [66]. Some form of ZMP regulation is used during the stance phase to prevent the foot from rolling. The obtained running gait is robot-like, has small flight times and with little ground clearance during flight. In the next section, we will explore dynamic running motions with non-trivial flight times.

2.2 Dynamic Running

While the ZMP based robots are quasi-statically stable with slow, robot-like motions, early dynamically stable running robots on the other hand employed the natural dynamics of the system through simple intuitive controllers to achieve life-like running gaits. Research on powered-legged robots that are dynamically stable began with Raibert’s groundbreaking work in the 1980’s [95]. Raibert pioneered the use of natural dynamics in the design and control of legged robots. He employed intuitive ideas to break up the control design into the regulation of three variables, the touchdown leg angle, body angle, and hopping height. A 3D hopper built by Raibert was able to hop, reaching top speeds of 2.2 m/s. Raibert’s controller structure has been used to demonstrate running gaits on two and four-legged robots with compliance by using a virtual leg approach, [53, 10]. Essentially, enhanced agility has been demonstrated on hopper-style robots (i.e., springy, prismatic leg) employing

intuitive controllers. These robots are highly underactuated, though for the most part, their control systems did not have to deal with stabilization of significant torso dynamics; indeed, if a torso was present, its center of mass was coincident with the hip joint [92]. Thus, the use of Raibert’s controllers to achieve stable running is possible on robots with favorable natural dynamics and appropriate morphology. However, a few complex legged robots have demonstrated running motions using Raibert’s controllers (see BigDog [96], Thumper [58].)

In continuation of a Raibert-style control design, Controlled Passive Dynamic Running (CPDR)-based controller [2] has been introduced by Buehler. The CPDR controller exploits the system’s passive dynamics by imposing desired trajectories via inverse dynamics to reduce energy spent for locomotion. The controller successfully demonstrated one-legged running on Monopod I [1], Monopod II [43, 2] at speeds of 1.2 m/s, 1.25 m/s respectively. However, the experimental system had a trivial torso, and a prismatic leg with series compliance.

Another robot that exhibited planar stable hopping, and could also hop over an unknown rough terrain with ground variations of up to 25% of the leg length of the robot is presented in [7]. The control strategy is based on active energy addition and removal, wherein, energy is removed from the system by leg actuation during leg compression and is added to the system during leg extension. This is done in a feedforward manner, as a function of time. Once again, the experimental system has a trivial torso, and a prismatic leg with series compliance, which is driven by a crank.

2.3 Passive Dynamics

The Raibert-style controller presented in the previous section employs favorable natural dynamics of the system to achieve running gaits. Taking the idea of using natural dynamics further, Tad McGeer inspired a new direction in bipedal robot design based on passive dynamics [78, 77]. McGeer drew inspiration from the Wright brothers, who first mastered the art of gliding (adding a propeller then became a minor technicality.) Passive dynamic machines are dynamically stable robots with no active feedback control or energy input aside from gravity, and walk or run on gentle slopes (see [25, 87]), with energy lost at each

foot collision being balanced by the conversion of potential energy to kinetic energy in going down a slope.

Detailed analysis of passive dynamic bipedal walking has been carried out by several researchers to demonstrate stable one-period gaits, period doubling phenomenon, and bifurcations leading to chaos [35, 40]. Experimental implementations of machines capable of walking and running down gentle slopes, with no power or actuation are presented in [78, 77].

Although passive dynamics based machines are energy efficient, however, in the absence of any control for making corrections, the constructed robot is not robust to perturbations, and in general cannot walk on flat ground². Furthermore, A significant amount of theoretical modeling and stability analysis is carried out before a passive dynamic based machine is even built.

2.4 Energy Efficient Locomotion

As discussed in the previous section, passive dynamics based robots are energy efficient, however, they tend to be very dependent on initial conditions and small variations in ground slope. As suggested by Kuo [72], energy efficiency is a chief requirement that needs to be satisfied for bipedal robots to be practical. Improved energy economy will provide legged robots with greater range, and independence, and provide a better ability to carry large loads or perform tasks for long durations of time. Thus, there is a need to improve energy efficiency of bipedal locomotion. The excellent article, [72], highlights the tradeoffs between performance and versatility in legged locomotion and examines the means by which energy economy can be enhanced in dynamic walking robots.

The efficiency of bipedal robots is being enhanced by using minimal actuation, incorporating compliance, or a combination of the two. Motivated by passive dynamic walkers, researchers have devised efficient means of walking on flat ground by injecting minimal amounts of energy at key points in the gait [26]. Kuo's detailed analysis in [71] of energet-

²A very recent article by Gomes and Ruina [39] suggests that they have found periodic collisionless motions of a walking model. Collisional losses are avoided by employing appropriate synchronized internal oscillations, such that the foot-strike velocity at impact is zero. This model is capable of walking on flat ground with non-zero, non infinitesimal speed and with zero energy input.

ics of walking, suggests that employing ankle push-off is more energetically efficient than employing hip torque on the stance leg. Another means of enhancing energy efficiency is by introducing compliant elements. The energetic benefits of springs in legged locomotion are well documented [3]. Springs can be used to store and release energy that otherwise would be lost as actuators do negative work, and springs can be used to isolate actuators from shocks arising from leg impacts with the ground. Although these benefits are more pronounced in running, compliance can also be used beneficially in walking [38, 62, 63]. Enhanced energy efficiency was shown using pneumatic artificial muscles in [120, 121, 117], using springs in series with motors in [93, 108], and using springs in parallel with motors in [136]. A combination of both methods, minimalistic actuation and compliant elements, is employed in the Cornell Biped [27], and the T.U. Delft bipeds TULip and Flame [52] in order to improve efficiency. The drawbacks of these highly efficient walkers are that they cannot lift their legs over obstacles, readily change speeds, or run.

2.5 Simple Models for Walking and Running

The previous section explored different means of enhancing energy efficiency for walking and running. Here we will see how the energy exchange between the kinetic and potential energies for walking and running will motivate simple models.

As put forth by Cavagna et al. [15], in walking, the center of mass is highest in mid-step, when the hip of the stance leg passes over the ankle. Thus in walking, changes in potential and kinetic energies *are out of phase*. In running, by comparison, the center of mass is lowest at mid-step, and changes in forward kinetic energy and gravitational potential energy are *in phase*. Remarkably, these basic energy transformation mechanisms exist in the gaits of birds, quadrupedal mammals, as well as humans [14]. In an effort to capture these extraordinary similarities, researchers proposed simple underlying models, whose purpose was to capture the dominant locomotion behavior resulting from complex, high-dimensional, nonlinear, dynamically coupled interactions between an organism and its environment.

For instance, it has been suggested that during walking the center of mass (COM) vaults over the stance leg analogous to an inverted pendulum [14], whereas in running, the

COM bounces analogous to bouncing on a pogo stick [34]. Engineers have encoded these observations in conceptual leg models such as the inverted pendulum (IP), consisting of a point mass atop a stiff massless rod, and the spring-loaded inverted pendulum (SLIP), consisting of a point mass atop a massless, compliant, prismatic leg; see [4, 34]. Further, the SLIP has been used to explain not only the basic dynamics of running, but that of walking as well [38]. This is due to the fact that a compliant leg is essential to incorporate the double support phase as a characteristic part of the walking motion.

These simple models have been extensively analyzed mathematically [69], and several controllers have been proposed to achieve stable walking [62] and running gaits [109] on these models. The SLIP has been used as a reference model for control design, and the designed controller is applied to complex systems [105, 106], whose dominant dynamics are those of the SLIP, with the intent of obtaining a closed-loop behavior that is predicted by the closed-loop SLIP model.

While the SLIP model is mathematically elegant and appears to describe the “center of mass” dynamics of many animals [9], it does not account for many features present in legged machines, namely legs that have non-zero mass, non-trivial torsos, and large rotating inertias typical of electric motors and gear drives [90].

Several extensions to the idealized SLIP model have been proposed to bring the model closer to reality. In TD-SLIP [8], a damper is introduced in addition to the compliance along with a torque driven hip. Rummel and Seyfarth introduce a segmented leg instead of the prismatic leg, and study the effect of compliance at the knee joint on the stability of the system. The SLIP has a point mass at the hip. Even if a torso is introduced, the COM of the torso is made coincident with the hip. Adding a torso with significant dynamics results in the ASLIP [61], where the COM of the torso is no longer coincident with the hip.

Next, looking ahead towards obtaining robust running, we look at how animals transverse unknown rough terrain with varying stiffness, and varying ground profiles, and see how these simple models can be used to explain robustness to ground variations. Biomechanics studies show that animals are able to robustly handle variations in ground height, and ground stiffness by varying their leg stiffness [31, 32, 29, 28]. Motivated by this, active force control has been suggested as a way to increase robustness to perturbations in ground

height and ground stiffness in [70]. Moreover, [32] suggests that incorporating an adjustable leg stiffness in the design of running robots is important if they are to match the agility and speed of animals on varied terrain.

2.6 Reduction-based Control Design

In the previous section, we have seen the SLIP serve as a reduced target model for a more complex anchor model [34]. The idea being that the dominant behavior of the complex system can be explained by the behavior of the simple model. Controllers designed on the target model then exhibit similar characteristics on the anchor.

On a similar note, and with the promise of analytical tractability, and numerical feasibility of lower-dimensional models, researchers have tried to find properties, whose invariance (possibly through feedback) leads to the reduction of a higher-dimensional model to a lower-dimensional model, wherein the behavior of the higher-dimensional complex model can be explained by the behavior of the lower-dimensional simpler model and vice versa. This results in control design which is based on reduction.

Spong and Bullo [114] introduce the concept of controlled symmetries, where the Lagrangian of the system is invariant under a particular group action. Considering the group action $SO(3)$ representing a ground slope change, it is shown that the kinetic energy and the impacts are invariant under this group action. The potential energy is then made invariant under this group action by using feedback control to achieve potential shaping, with the end result being the invariance of the Lagrangian. With this, a passive walking gait can be made to walk on a ground of any slope using active control. This approach requires a fully actuated biped.

Ames et. al. [6] utilize controlled symmetries and introduce the concept of functional Routhian reduction, where conserved quantities are functions of cyclic variables rather than constants. Extending the functional Routhian reduction to a hybrid setting, a geometric reduction can be achieved. This is applied to a fully actuated biped to achieve stable walking gaits. Gregg and Spong [42] present controlled reduction, a variant of functional Routhian reduction, to decouple a 3D biped's sagittal-plane motion from the yaw and lean modes.

The tutorial paper [47] develops a comprehensive hybrid model of a 3D biped and presents control designs based on hybrid zero dynamics and functional Routhian reduction.

The next section introduces hybrid zero dynamics, which is a form of dynamic reduction through feedback control, where the feedback imposes virtual constraints and enables studying the stability properties of a higher-dimensional model by studying the stability properties of a restricted (that is, respecting the virtual constraints) lower-dimensional model.

2.7 Hybrid Zero Dynamics

Ground-breaking work on formal stability analysis of bipedal locomotion began with research on RABBIT leading to [46], where systems with impulse effects are used to characterize the hybrid dynamics of bipedal walking, and the method of Poincaré is employed to reduce the stability of a limit cycle corresponding to a walking gait to the stability of a discrete map. The bipedal robot RABBIT was planar, had revolute knees, and a non-trivial torso [16]. It was deliberately designed to have point feet in order to inspire new analytical control approaches to stabilizing periodic motion in underactuated mechanical systems, and hence move beyond flat-footed walking gaits. This research gave rise to the methods of virtual constraints and hybrid zero dynamics [46, 131, 134, 16, 82, 130]. Virtual constraints are holonomic constraints on a robot’s configuration that are asymptotically achieved through feedback control and are used to synchronize the evolution of the various links throughout a stride. Virtual constraints lead to a family of feedback laws that accomplishes two things: (i) the creation of a low-dimensional surface that is hybrid invariant (i.e., invariant under both the closed-loop continuous dynamics and the impact map); and (ii) the surface is rendered exponentially attractive in the closed-loop hybrid system. The restriction dynamics associated with the invariant surface is called the Hybrid Zero Dynamics (HZD). The HZD is used as a means of stripping away as much of the complexity of dynamic locomotion problems as possible in order to find the *essential dynamics* of walking and running. The initial work applied to robots with rigid links and one degree of underactuation, the canonical example being a planar bipedal robot with $N \geq 2$ rigid links, $N - 1$ independent actuators,

and the legs terminated in passive point feet; fully-actuated robots (i.e., feet with actuated ankles) have also been studied [23, 108, 18]. Hybrid zero dynamics based controllers have also been employed to theoretically design stable gaits for running on planar bipeds [21, 85], walking on 3D (spatial) bipeds [48, 19], and steering of a 3D bipedal robot [20].

A related approach based on designing a linear feedback controller that stabilizes the time-varying transverse linearization of a hybrid system along a periodic orbit has been developed in [75, 111, 110, 112].

2.8 Compliant Hybrid Zero Dynamics

Using the theory of hybrid zero dynamics, RABBIT was established as a very successful walker [132, 103]. However steady state running was never achieved. In [85], it was conjectured that this was due to actuator saturation: upon transition from the flight phase to the stance phase, the actuators in the stance leg had to perform large amounts of negative work to decelerate the robot’s center of mass, and then do positive work to redirect it upward for the subsequent flight phase. This motivated the inclusion of a spring in MABEL so that much of this work could be done passively. Further analysis in [92] suggests that including compliance in the leg length direction has an additional benefit: it can create more favorable unilateral ground contact conditions to avoid slipping.

The presence of compliance has led to new control challenges that cannot be met with the initial theory developed for RABBIT. On the mathematical side, compliance increases the degree of underactuation, which in turn makes it more difficult to meet the invariance condition required for a hybrid zero dynamics to exist. This technical difficulty was overcome in [84] with a technique called a “deadbeat hybrid extension”.

A second challenge arising from compliance is how to use it effectively. A first attempt in [83] at designing a controller for a biped with springs took advantage of the compliance along a steady state walking gait, but “fought it” during transients; the compliance was effectively canceled in the HZD (for details, see [92, p. 1790]). The problem of ensuring that the feedback action preserves the compliant nature of the system was studied in [92, 91, 90] for the task of hopping in a monopod, where the HZD itself was designed to be compliant,

resulting in compliant hybrid zero dynamics.

2.9 Summary

The survey reported in this chapter sets the stage for the thesis. MABEL was designed with compliance to primarily enable it to run. Drawing from the literature survey just presented, we can make the following observations. (a) Compliance can also be used for walking, (b) Compliance can be effectively utilized for improving energy efficiency of the walking gait, (c) Hybrid zero dynamics based controllers can be formulated to preserve the natural compliance of the system as part of the restricted dynamics, and achieve stable walking and running, and (d) If we can vary the leg stiffness, essentially the compliance in the restricted dynamics, then we can obtain running gaits that are robust to ground variations and ground stiffness. The end result would then be stable, efficient walking, and stable, robust running.

The work presented in the thesis shows how this is achieved. Chapter IV extends the compliant HZD based control design to achieve stable, efficient, and fast walking. Chapter VI presents a control design for embedding active force control within the compliant HZD framework to enable dynamically varying the effective leg stiffness to achieve stable and running.

CHAPTER III

Control-Oriented Model of MABEL

This chapter presents details about the morphology of MABEL, and develops the appropriate mathematical models for the study of walking and running.

3.1 Description of MABEL

MABEL is a planar bipedal robot comprised of five links assembled to form a torso and two legs with knees; see Figure 1.1. The robot weighs 65 kg, is 1 m at the hip, and mounted on a boom of radius 2.25 m. The legs are terminated in point feet. All actuators are located in the torso, so that the legs are kept as light as possible; this is to facilitate rapid leg swinging for running. Unlike most bipedal robots, the actuated degrees of freedom of each leg do not correspond to the knee and hip angles. Instead, for each leg, a collection of cable-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 consisting of the line connecting the hip to the toe, and the second motor is connected in series with a spring in order to control the length or shape of the virtual leg; see Figure 3.1. The reader is referred to [50, 49, 58] for more details on the transmission.

The springs in MABEL serve to isolate the reflected rotor inertia of the leg-shape motors from the impact forces at leg touchdown and to store energy in 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, when both legs are off the ground. These

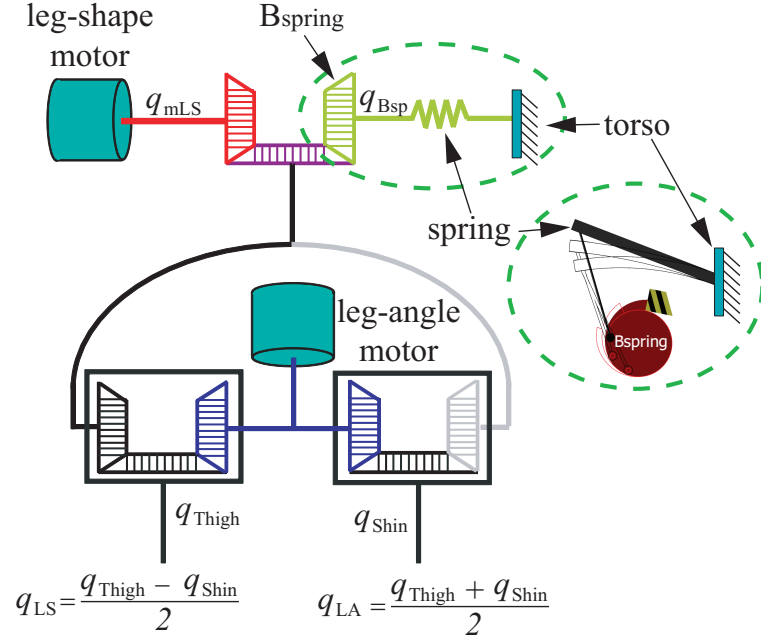


Figure 3.1: MABEL’s powertrain (courtesy of Hae-Won Park.) The powertrain is comprised of three cable differentials (same for each leg), all housed in the torso. Two motors and a spring are connected to the traditional hip and knee joints via three differentials. On the robot, the differentials are realized via cables and pulleys [58] and not via gears. They are connected such that the actuated variables are leg angle and leg shape, see Figure 1.1, and so that the spring is in series with the leg shape motor. The base of the spring is grounded to the torso and the other end is connected to the pulley B_{spring} via a cable, which makes the spring *unilateral*. When the spring reaches its rest length, the pulley hits a hard stop, formed by a very stiff damper. When this happens, the leg shape motor is, for all intents and purposes, rigidly connected to leg shape through a gear ratio.

properties (shock isolation and energy storage) enhance the energy efficiency of running and reduce the overall actuator power requirements. This is also true for walking as we will demonstrate experimentally. MABEL has a unilateral spring which compresses but does not extend beyond its rest length. This ensures that springs are present when they are useful for shock attenuation and energy storage, and absent when they would be a hindrance for lifting the legs from the ground.

3.2 MABEL Model

A hybrid model appropriate for a walking gait, comprised of a continuous single support phase and an instantaneous double support phase, is developed next. The impact model at double support is based on [57]. The single support model is a pinned, planar, 5-link kinematic chain with revolute joints and rigid links. Because the compliance is unilateral, it will be more convenient to model it as an external force when computing the Lagrangian, instead of including it as part of the potential energy.

3.2.1 MABEL's Unconstrained Dynamics

The configuration space Q_e of the unconstrained dynamics of MABEL is a simply-connected subset of $\mathbb{S}^7 \times \mathbb{R}^2$: five DOF are associated with the links in the robot's body, two DOF are associated with the springs in series with the two leg-shape motors, and two DOF are associated with the horizontal and vertical position of the robot in the sagittal plane. A set of coordinates suitable for parametrization of the robot's linkage and transmission is $q_e := (q_{LA_{st}}; q_{mLS_{st}}; q_{Bsp_{st}}; q_{LA_{sw}}; q_{mLS_{sw}}; q_{Bsp_{sw}}; q_{Tor}; p_{hip}^h; p_{hip}^v)$, the subscripts st and sw refer to the stance and swing legs respectively. As in Figure 1.1 and Figure 3.1, q_{Tor} is the torso angle, and $q_{LA_{st}}$, $q_{mLS_{st}}$, and $q_{Bsp_{st}}$ are the leg angle, leg-shape motor position, and B_{spring} position, respectively for the stance leg. The swing leg variables, $q_{LA_{sw}}$, $q_{mLS_{sw}}$ and $q_{Bsp_{sw}}$ are defined similarly. For each leg, q_{LS} is determined from q_{mLS} and q_{Bsp} by

$$q_{LS} = 0.0318q_{mLS} + 0.193q_{Bsp}. \quad (3.1)$$

This reflects the fact that the cable differentials place the spring in series with the motor, with the pulleys introducing a gear ratio. The coordinates p_{hip}^h, p_{hip}^v are the horizontal and vertical positions of the hip in the sagittal plane. The hip position is chosen as an independent coordinate instead of the center of mass because it was observed that this choice significantly reduces the number of terms in the symbolic expressions for the dynamics.

The equations of motion are obtained using the method of Lagrange. The Lagrangian

for the unconstrained system, $\mathcal{L}_e : TQ_e \rightarrow \mathbb{R}$ is defined by

$$\mathcal{L}_e = \mathcal{K}_e - \mathcal{V}_e, \quad (3.2)$$

where, $\mathcal{K}_e : TQ_e \rightarrow \mathbb{R}$ and $\mathcal{V}_e : Q_e \rightarrow \mathbb{R}$ 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}_e^{link} , the kinetic energy of the stance and swing leg transmissions, $\mathcal{K}_e^{trans_{st}}$, $\mathcal{K}_e^{trans_{sw}}$, and the kinetic energy of the boom, \mathcal{K}_e^{boom} ,

$$\begin{aligned} \mathcal{K}_e(q_e, \dot{q}_e) = & \mathcal{K}_e^{link}(q_e, \dot{q}_e) + \mathcal{K}_e^{trans_{st}}(q_e, \dot{q}_e) + \\ & \mathcal{K}_e^{trans_{sw}}(q_e, \dot{q}_e) + \mathcal{K}_e^{boom}(q_e, \dot{q}_e). \end{aligned} \quad (3.3)$$

The linkage model is standard. Physically, the boom constrains 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, as in [129, p. 94], 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 [44].

Similar notation is used for the potential energy,

$$\begin{aligned} \mathcal{V}_e(q_e) = & \mathcal{V}_e^{link}(q_e) + \mathcal{V}_e^{trans_{st}}(q_e) + \\ & \mathcal{V}_e^{trans_{sw}}(q_e) + \mathcal{V}_e^{boom}. \end{aligned} \quad (3.4)$$

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 unconstrained robot dynamics can be determined through Lagrange's equations

$$\frac{d}{dt} \frac{\partial \mathcal{L}_e}{\partial \dot{q}_e} - \frac{\partial \mathcal{L}_e}{\partial q_e} = \Gamma_e, \quad (3.5)$$

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

$$\begin{aligned} \Gamma_e = & B_e u + E_{\text{ext}}(q_e) F_{\text{ext}} + \\ & B_{\text{fric}} \tau_{\text{fric}}(q_e, \dot{q}_e) + B_{\text{sp}} \tau_{\text{sp}}(q_e, \dot{q}_e), \end{aligned} \quad (3.6)$$

where the matrices B_e , 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) and (3.4), respectively, results in the second-order dynamical model

$$D_e(q_e) \ddot{q}_e + C_e(q_e, \dot{q}_e) \dot{q}_e + G_e(q_e) = \Gamma_e \quad (3.7)$$

for the unconstrained dynamics of MABEL. Here D_e is the inertia matrix, the matrix C_e contains Coriolis and centrifugal terms, and G_e is the gravity vector.

3.2.2 Dynamics of Stance

For modeling the stance phase, the stance toe is assumed to act as a passive pivot joint (no slip, no rebound and no actuation). Hence, the Cartesian position of the hip, $(p_{\text{hip}}^h, p_{\text{hip}}^v)$, is defined by the coordinates of the stance leg and torso. 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, $q_{\text{Bsp}_{\text{sw}}} \equiv 0$. It follows from (3.1) that $q_{\text{mLS}_{\text{sw}}}$ and $q_{\text{LS}_{\text{sw}}}$ are related by a gear ratio; $q_{\text{mLS}_{\text{sw}}}$ is taken as the independent variable. With these assumptions, the generalized configuration variables in stance are taken as $q_s := (q_{\text{LA}_{\text{st}}}; q_{\text{mLS}_{\text{st}}}; q_{\text{Bsp}_{\text{st}}}; q_{\text{LA}_{\text{sw}}}; q_{\text{mLS}_{\text{sw}}}; q_{\text{Tor}})$.

The stance dynamics is obtained by applying the above holonomic constraints to the model of Section 3.2.1. The stance configuration space is therefore a co-dimension three submanifold of Q_e , i.e., $Q_s := \{q_e \in Q_e \mid q_{\text{Bsp}_{\text{sw}}} \equiv 0, p_{\text{toe}_{\text{st}}}^h \equiv 0, p_{\text{toe}_{\text{st}}}^v \equiv 0\}$. For later use, we denote by

$$q_e = \Upsilon_s(q_s) \quad (3.8)$$

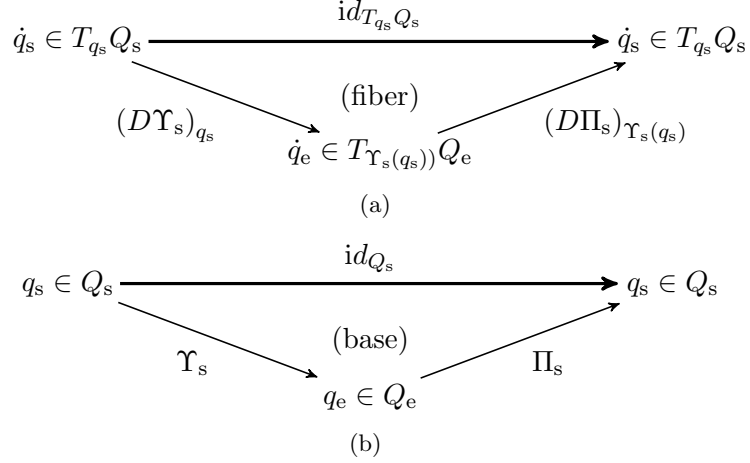


Figure 3.2: Stance commutative diagrams

the value of q_e when $q_s \in Q_s$, and by

$$q_s = \Pi_s(q_e) \quad (3.9)$$

the value of q_e projected onto $Q_s \subset Q_e$, such that, $\Pi_s \circ \Upsilon_s = id_{Q_s}$ as suggested by the commutative diagram of Figure 3.2. Further, the unconstrained velocity \dot{q}_e can be obtained from the stance velocity \dot{q}_s through the differential of the map Υ_s at the point $q_s \in Q_s$, i.e.,

$$\dot{q}_e = (D\Upsilon_s)_{q_s}(\dot{q}_s), \quad (3.10)$$

where $(D\Upsilon_s)_{q_s} : T_{q_s} Q_s \rightarrow T_{\Upsilon_s(q_s)} Q_e$. Similarly, the stance velocity can be obtained from the unconstrained velocity through the differential of the map Π_s at the point $q_e \in Q_e$, i.e.,

$$\dot{q}_s = (D\Pi_s)_{q_e}(\dot{q}_e), \quad (3.11)$$

where $(D\Pi_s)_{q_e} : T_{q_e} Q_e \rightarrow T_{\Pi_s(q_e)} Q_s$. Moreover, $(D\Pi_s)_{\Upsilon_s(q_s)} \circ (D\Upsilon_s)_{q_s} = id_{T_{q_s} Q_s}$.

The resulting constrained Lagrangian $\mathcal{L}_s : TQ_s \rightarrow \mathbb{R}$ can be expressed as

$$\mathcal{L}_s := \mathcal{L}_e(q_e, \dot{q}_e) \Big|_{\{q_{\text{Bsp}_{\text{sw}}} \equiv 0, p_{\text{toe}_{\text{st}}}^h \equiv 0, p_{\text{toe}_{\text{st}}}^v \equiv 0\}}, \quad (3.12)$$

and the dynamics of stance are obtained through Lagrange's equations, expressed in stan-

standard form as

$$D_s(q_s) \ddot{q}_s + C_s(q_s, \dot{q}_s) \dot{q}_s + G_s(q_s) = \Gamma_s, \quad (3.13)$$

where, $\Gamma_s := B_s u + B_{fric} \tau_{fric}(q_s, \dot{q}_s) + B_{sp} \tau_{sp}(q_s, \dot{q}_s)$ is the vector of generalized forces acting on the robot.

The state-space form of the stance dynamics, with the state vector $x_s := (q_s; \dot{q}_s) \in TQ_s$, can be expressed as,

$$\begin{aligned} \dot{x}_s &:= \begin{bmatrix} \dot{q}_s \\ \ddot{q}_s \end{bmatrix} = \begin{bmatrix} \dot{q}_s \\ -D_s^{-1} H_s \end{bmatrix} + \begin{bmatrix} 0 \\ D_s^{-1} B_s \end{bmatrix} u \\ &=: f_s(x_s) + g_s(x_s) u, \end{aligned} \quad (3.14)$$

where, f_s, g_s are the drift and input vector fields for the stance dynamics, and $H_s := C_s(q_s, \dot{q}_s) \dot{q}_s + G_s(q_s) - B_{fric} \tau_{fric}(q_s, \dot{q}_s) - B_{sp} \tau_{sp}(q_s, \dot{q}_s)$.

3.2.3 Stance to Stance Transition Map

An impact occurs when the swing leg touches the ground, modeled here as an inelastic contact between two rigid bodies. In addition to modeling the impact of the leg with the ground and the associated discontinuity in the generalized velocities of the robot as in [57], the transition map accounts for the assumption that the spring on the swing leg is at its rest length, and for the relabeling of the robot's coordinates so that only one stance model is necessary. In particular, the transition map consists of three subphases executed in the following order: (a) standard rigid impact model [57]; (b) adjustment of spring rest length in the new swing leg; and (c) coordinate relabeling.

Before entering into the details, the spring is discussed. To meet our modeling assumption of Section 3.2.2, the post-transition spring position on the new swing leg has to be non-deflected. This requirement makes the pre and post-transition position coordinates not identical. Physically, the spring being non-deflected is a well-founded assumption because as soon as weight of the robot comes off the former stance leg, the spring rapidly relaxes and the pulley q_{Bsp} comes to rest on the hard stop. This causes a change in torque on the leg-shape motor, and either the motor shaft or the leg shape needs to reposition to maintain a balance of torques in the leg shape differentials. Because the leg shape has a

high reflected inertia at the motor, it is the motor that repositions. Further, since q_{LS} is a linear combination of q_{mLS} and q_{Bsp} per (3.1), we can assume the spring and motor position change appropriately such that the linkage positions q_{LS}^+ , q_{LS}^- are still identical. Thus, the pre and post-transition linkage coordinates still remain identical.

The robot physically transitions from one stance phase to the next when the swing toe contacts the ground. It is assumed that there is no rebound or slip at impact, and that the old stance leg lifts off from the ground without interaction. The external forces are represented by impulses, and since the actuators cannot generate impulses, they are ignored during impact. Mathematically, the transition then occurs when the solution of (3.14) intersects the co-dimension one switching manifold defined by the zero level set of the threshold function $H_{s \rightarrow s} : TQ_s \rightarrow \mathbb{R}$,

$$\mathcal{S}_{s \rightarrow s} := \{x_s \in TQ_s \mid H_{s \rightarrow s}(x_s) = 0\}, \quad (3.15)$$

with $H_{s \rightarrow s}(x_s) = p_{\text{toesw}}^v$, where p_{toesw}^v is the vertical position of the swing toe.

The stance to stance transition map, $\Delta_{s \rightarrow s} : \mathcal{S}_{s \rightarrow s} \rightarrow TQ_s$, is defined as

$$\Delta_{s \rightarrow s}(x_s^-) := \begin{bmatrix} \Delta_{s \rightarrow s}^q(q_s^-) \\ (\Delta_{s \rightarrow s}^{\dot{q}})_{q_s^-}(\dot{q}_s^-) \end{bmatrix}, \quad (3.16)$$

where, $x_s^- = (q_s^-; \dot{q}_s^-) \in \mathcal{S}_{s \rightarrow s}$ is the final state of the stance phase and the base and fiber components, $\Delta_{s \rightarrow s}^q : Q_s \rightarrow Q_s$, $(\Delta_{s \rightarrow s}^{\dot{q}})_{q_s^-} : T_{q_s}Q_s \rightarrow T_{\Delta_{s \rightarrow s}^q(q_s^-)}Q_s$ define the transition maps for the configuration variables and their velocities, respectively. The initial state of the stance phase, $x_s^+ \in TQ_s$, is the post impact state and is obtained as,

$$x_s^+ = \Delta_{s \rightarrow s}(x_s^-). \quad (3.17)$$

The impacts being modeled here are those of the swing foot with the ground and that of the stance spring hitting the hard stop when the stance leg comes off the ground. Here we model both these impacts to occur at precisely the same instant¹. Mathematically, the

¹We have checked that first doing the standard impact for the swing leg, and then doing a second impact

$$\begin{array}{ccc}
\dot{q}_s^- \in T_{q_s^-} Q_s & \xrightarrow{(\Delta_{s \rightarrow s}^{\dot{q}})_{q_s^-}} & \dot{q}_s^+ \in T_{\Delta_{s \rightarrow s}^q(q_s^-)} Q_s \\
(D\Upsilon_s)_{q_s^-} \downarrow & \text{(fiber)} & \uparrow (D\Pi_s)_{\Delta_{GndStp}^q \circ \Upsilon_s(q_s^-)} \circ R \\
\dot{q}_e^- \in T_{\Upsilon_s(q_s^-)} Q_e & \xrightarrow{(\Delta_{GndStp}^{\dot{q}})_{\Upsilon_s(q_s^-)}} & \dot{q}_e^+ \in T_{\Delta_{GndStp}^q \circ \Upsilon_s(q_s^-)} Q_e
\end{array}
\tag{a}$$

$$\begin{array}{ccc}
q_s^- \in Q_s & \xrightarrow{\Delta_{s \rightarrow s}^q} & q_s^+ \in Q_s \\
\Upsilon_s \downarrow & \text{(base)} & \uparrow \Pi_s \circ R \\
q_e^- \in Q_e & \xrightarrow{\Delta_{GndStp}^q = \Pi_{Bsp}} & q_e^+ \in Q_e
\end{array}
\tag{b}$$

Figure 3.3: Stance to Stance commutative diagrams

simultaneous impact with the ground and the impact with the hard stop are abstracted by the impact map $\Delta_{GndStp} : TQ_e \rightarrow TQ_e$. The base and fiber components of the stance to stance transition map can then be expressed using the impact map as,

$$\Delta_{s \rightarrow s}^q = \Pi_s \circ R \circ \Delta_{GndStp}^q \circ \Upsilon_s, \tag{3.18}$$

$$(\Delta_{s \rightarrow s}^{\dot{q}})_{q_s^-} = (D\Pi_s)_{\Delta_{GndStp}^q \circ \Upsilon_s(q_s^-)} \circ R \circ (\Delta_{GndStp}^{\dot{q}})_{\Upsilon_s(q_s^-)} \circ (D\Upsilon_s)_{q_s^-}, \tag{3.19}$$

such that diagram of Figure 3.3 commutes. Υ_s , Π_s are as in (3.8), (3.9) respectively, and R is a linear operator representing coordinate relabeling as found in [130, p. 57].

The rest of this section will focus on deriving the base and fiber components of the impact map Δ_{GndStp} .

As per earlier discussions, the linkage positions (q_{LA} , q_{LS} on either leg, and q_{Tor}) are invariant with respect to an impact with the ground. However, the impact of the pulley B_{spring} with the hard stop requires a change in the position of the transmission variable (specifically $q_{mLS_{st}}$) such that the linkage positions are invariant. Thus the impact map for

for q_{Bsp} hitting the hard stop, with the constraint that the new stance leg end velocity remains zero, gives the same result as the model presented here.

the coordinates can be expressed as

$$\Delta_{GndStp}^q := \Pi_{Bsp}. \quad (3.20)$$

Π_{Bsp} is a projection from Q_e onto the co-dimension two submanifold $\{q_e \in Q_e \mid q_{Bsp_{st}} \equiv 0, q_{Bsp_{sw}} \equiv 0\}$ such that the linkage coordinates $(q_{LA}, q_{LS}, q_{Tor})$ remain invariant under the projection. Thus Π_{Bsp} resets the spring to its rest position by modifying the leg-shape motor position such that the leg-shape position itself is unchanged.

Next, the impact map for the velocities is derived as follows. Let I_R be the impulsive force on the foot due to the ground-foot impact and let τ_R be the impulsive torque on the spring due to pulley B_{spring} hitting the hard stop. Then the generalized external impulsive force acting on the system is obtained from the principle of virtual work as,

$$F_{ext} = \left(\frac{\partial p_{toesw}}{\partial q_e} \right)^T I_R + \left(\frac{\partial q_{Bsp_{st}}}{\partial q_e} \right)^T \tau_R. \quad (3.21)$$

We have three constraints that need to be satisfied at impact. The first condition is for the new swing leg to have zero spring velocity. The second condition is for the new stance toe to have zero velocity. The third constraint is obtained by integrating the unconstrained dynamics, (3.7), over the duration of the instantaneous event. These conditions are

$$\dot{q}_{Bsp_{st}}^+ = 0 \implies \frac{\partial q_{Bsp_{st}}}{\partial q_e} \dot{q}_e^+ = 0, \quad (3.22)$$

$$\dot{p}_{toesw}^+ = 0 \implies \frac{\partial p_{toesw}}{\partial q_e} \dot{q}_e^+ = 0, \quad (3.23)$$

$$D_e(q_e^+) \dot{q}_e^+ - D_e(q_e^-) \dot{q}_e^- = F_{ext}. \quad (3.24)$$

By assembling the constraints (3.22)-(3.24), and solving for the post-impact velocity, we

obtain the map,

$$\begin{aligned} \left(\Delta_{GndStp}^{\dot{q}} \right)_{q_e^-} (\dot{q}_e^-) &= \begin{bmatrix} I & 0 & 0 \end{bmatrix} \\ &\begin{bmatrix} \frac{\partial q_{Bspst}}{\partial q_e} & 0 & 0 \\ \frac{\partial p_{toesw}}{\partial q_e} & 0 & 0 \\ D_e \left(\Delta_{GndStp}^q(q_e^-) \right) & -\frac{\partial p_{toesw}}{\partial q_e}^T & -\frac{\partial q_{Bspst}}{\partial q_e}^T \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ D_e(q_e^-) \dot{q}_e^- \end{bmatrix}, \end{aligned} \quad (3.25)$$

where Δ_{GndStp}^q is as defined in (3.20). With this, the base and fiber components of the stance to stance transition map, (3.16), are completely defined.

3.2.4 Hybrid Model of Walking

The hybrid model of walking is based on the dynamics developed in Section 3.2.2 and transition map derived in Section 3.2.3. The continuous dynamics with discrete state transitions is represented as,

$$\Sigma_w : \begin{cases} \mathcal{X}_s = TQ_s \\ \mathcal{S}_{s \rightarrow s} = \{x_s \in \mathcal{X}_s \mid H_{s \rightarrow s}(x_s) = 0\} \\ \dot{x}_s = f_s(x_s) + g_s(x_s)u, & x_s^- \notin \mathcal{S}_{s \rightarrow s} \\ x_s^+ = \Delta_{s \rightarrow s}(x_s^-), & x_s^- \in \mathcal{S}_{s \rightarrow s}. \end{cases} \quad (3.26)$$

3.2.5 Dynamics of Flight

In the flight phase, both the feet are off the ground, and the robot follows a ballistic motion under the influence of gravity. Thus the flight dynamics can be modeled by the unconstrained dynamics developed in the previous section. Further, for reasons mentioned for the swing leg during the stance phase, and the fact that neither leg is in contact with the ground during the flight phase, it will be assumed that the springs on each leg do not deflect during the flight phase. Therefore, $q_{Bspst} \equiv 0, q_{Bspsw} \equiv 0$. This assumption is computationally advantageous since it eliminates the stiffness in the model while integrating the differential equations. Thus, the configuration space of the flight dynamics is a co-dimension two submanifold of Q_e , i.e., $Q_f := \{q_e \in Q_e \mid q_{Bspst} \equiv 0, q_{Bspsw} \equiv 0\}$. It

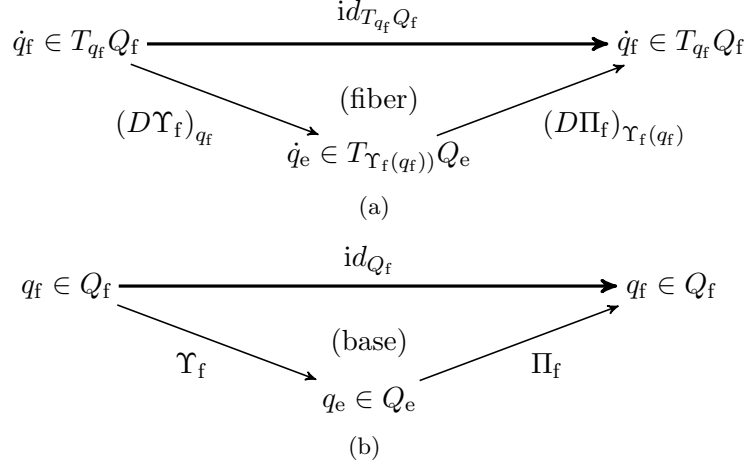


Figure 3.4: Flight commutative diagrams

follows that, the generalized configuration variables in the flight phase can be taken as $q_f := \left(q_{LA_{st}}; q_{mLS_{st}}; q_{LA_{sw}}; q_{mLS_{sw}}; q_{Tor}; p_{hip}^h; p_{hip}^v \right)$. For later use, we denote by

$$q_e = \Upsilon_f(q_f), \quad (3.27)$$

the value of q_e when $q_f \in Q_f$, and

$$q_f = \Pi_f(q_e), \quad (3.28)$$

the value of q_e projected onto $Q_f \subset Q_e$, such that, $\Pi_f \circ \Upsilon_f = id_{Q_f}$ as suggested by the commutative diagram of Figure 3.4. Further, the unconstrained velocity \dot{q}_e can be obtained from the flight velocity \dot{q}_f through the differential of the map Υ_f at the point $q_f \in Q_f$, i.e.,

$$\dot{q}_e = (D\Upsilon_f)_{q_f}(\dot{q}_f), \quad (3.29)$$

where $(D\Upsilon_f)_{q_f} : T_{q_f} Q_f \rightarrow T_{\Upsilon_f(q_f)} Q_e$. Similarly, the stance velocity can be obtained from the unconstrained velocity through the differential of the map Π_f at the point $q_e \in Q_e$, i.e.,

$$\dot{q}_f = (D\Pi_f)_{q_e}(\dot{q}_e), \quad (3.30)$$

where $(D\Pi_f)_{q_e} : T_{q_e} Q_e \rightarrow T_{\Pi_f(q_e)} Q_f$. Moreover, $(D\Pi_f)_{\Upsilon_f(q_f)} \circ (D\Upsilon_f)_{q_f} = id_{T_{q_f} Q_f}$.

Thus, the resulting Lagrangian $\mathcal{L}_f : TQ_f \rightarrow \mathbb{R}$ in the flight phase can be expressed as

$$\mathcal{L}_f(q_f, \dot{q}_f) = \mathcal{L}_e(q_e, \dot{q}_e) \Big|_{q_{Bsp_{st}} \equiv 0, q_{Bsp_{sw}} \equiv 0}, \quad (3.31)$$

and the flight dynamics can be expressed in the standard form as

$$D_f(q_f) \ddot{q}_f + C_f(q_f, \dot{q}_f) \dot{q}_f + G_f(q_f) = \Gamma_f, \quad (3.32)$$

where, $\Gamma_f := B_f u + B_{fric} \tau_{fric}(q_f, \dot{q}_f) + B_{sp} \tau_{sp}(q_f, \dot{q}_f)$ is the vector of generalized forces acting on the robot.

The state-space form of the flight dynamics, with the state vector $x_f := (q_f; \dot{q}_f) \in TQ_f$, can be expressed as,

$$\begin{aligned} \dot{x}_f &:= \begin{bmatrix} \dot{q}_f \\ \ddot{q}_f \end{bmatrix} = \begin{bmatrix} \dot{q}_f \\ -D_f^{-1} H_f \end{bmatrix} + \begin{bmatrix} 0 \\ D_f^{-1} B_f \end{bmatrix} u \\ &=: f_f(x_f) + g_f(x_f) u \end{aligned} \quad (3.33)$$

where, f_f, g_f are the drift and input vector fields for the flight dynamics, and $H_f = C_f(q_f, \dot{q}_f) \dot{q}_f + G_f(q_f) - B_{fric} \tau_{fric}(q_f, \dot{q}_f) - B_{sp} \tau_{sp}(q_f, \dot{q}_f)$.

3.2.6 Stance to Flight Transition Map

Physically, the robot takes off when the normal component of the ground reaction force acting on the stance toe, $F_{toe_{st}}^N$, becomes zero. The ground reaction force at the stance toe can be computed as a function of the acceleration of the COM and thus depends on the inputs $u \in \mathcal{U}$ of the system described by (3.14). To formally express the takeoff event, we first define a trivial fiber bundle,

$$\pi : B \rightarrow TQ_s, \quad (3.34)$$

where $B = TQ_s \times \mathcal{U}$. Mathematically, takeoff occurs when the solution of (3.14) intersects the co-dimension one switching manifold $\mathcal{S}_{s \rightarrow f}$ in the fiber bundle (3.34), defined as,

$$\mathcal{S}_{s \rightarrow f} := \{(x_s, u) \in TQ_s \times \mathcal{U} \mid H_{s \rightarrow f}(x_s, u) = 0\}, \quad (3.35)$$

where the threshold function $H_{s \rightarrow f} : TQ_s \times \mathcal{U} \rightarrow \mathbb{R}$ is defined as $H_{s \rightarrow f}(x_s, u) = F_{toest}^N$, with F_{toest}^N being the normal component of the ground reaction force at the stance foot.

The stance to flight transition map, $\Delta_{s \rightarrow f} : \mathcal{S}_{s \rightarrow f} \rightarrow TQ_f$, is defined as

$$\Delta_{s \rightarrow f}(x_s^-, u^-) := \begin{bmatrix} \Delta_{s \rightarrow f}^q(q_s^-) \\ (\Delta_{s \rightarrow f}^{\dot{q}})_{q_s^-}(\dot{q}_s^-, u^-) \end{bmatrix}, \quad (3.36)$$

where, $x_s^- = (q_s^-; \dot{q}_s^-) \in TQ_s$ is the final state of the stance phase and $u^- \in \mathcal{U}$ is the input at this instant. The base and fiber components, $\Delta_{s \rightarrow f}^q : Q_s \rightarrow Q_f$, $(\Delta_{s \rightarrow f}^{\dot{q}})_{q_s^-} : T_{q_s}Q_s \times \mathcal{U} \rightarrow T_{\Delta_{s \rightarrow f}^q(q_s)}Q_f$ define the transition maps for the configuration variables and their velocities, respectively. The initial state of the flight phase, $x_f^+ \in TQ_f$, is the post transition state and is obtained as,

$$x_f^+ = \Delta_{s \rightarrow f}(x_s^-, u^-). \quad (3.37)$$

On transition from the stance to flight phase, the stance leg comes off the ground and takeoff occurs. During the stance phase, the spring is compressed. When the stance leg comes off the ground, the spring rapidly decompresses until it reaches its rest position. At this instant, there is an impact of the pulley B_{spring} hitting the hard stop. Mathematically, this is captured by the impact map $\Delta_{Stp} : TQ_e \rightarrow TQ_e$ representing the impact with the hard stop. The base and fiber components of the stance to flight transition map can then be expressed using the impact map as,

$$\Delta_{s \rightarrow f}^q = \Pi_f \circ \Delta_{Stp}^q \circ \Upsilon_s, \quad (3.38)$$

$$(\Delta_{s \rightarrow f}^{\dot{q}})_{q_s^-} = (D\Pi_f)_{\Delta_{Stp}^q \circ \Upsilon_s(q_s^-)} \circ (\Delta_{Stp}^{\dot{q}})_{\Upsilon_s(q_s^-)} \circ (D\Upsilon_s)_{q_s^-} \circ \pi, \quad (3.39)$$

such that diagram of Figure 3.5 commutes. Υ_s , Π_f are as in (3.8), (3.28) respectively, and π is as defined in (3.34).

The rest of this section will focus on deriving the base and fiber components of the impact map Δ_{Stp} .

As per earlier discussions, the impact of the pulley B_{spring} with the hard stop requires a change in the position of the transmission variable (specifically q_{mLSst}) such that the linkage

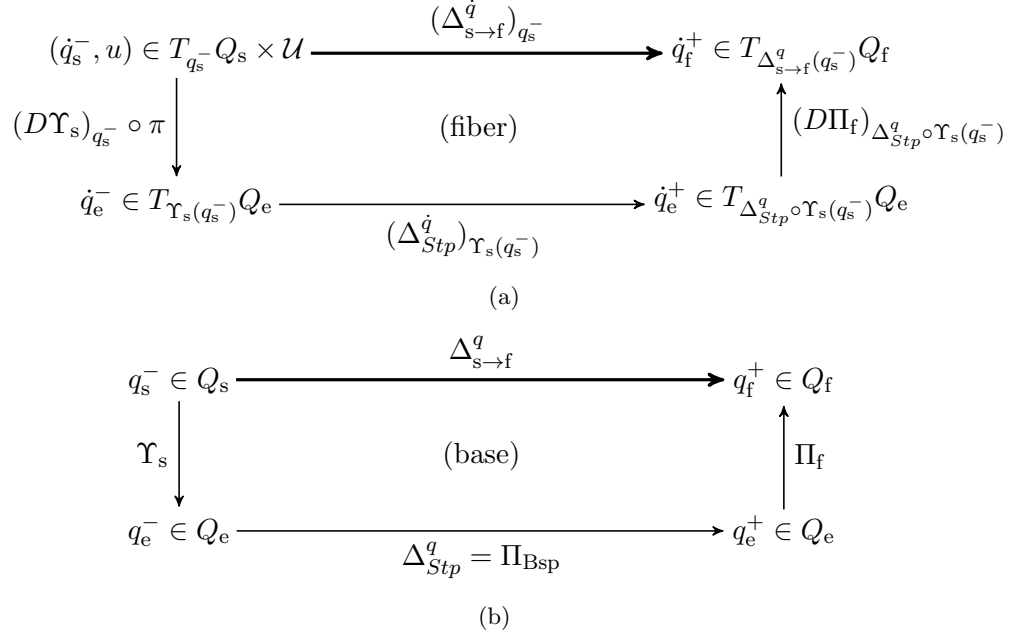


Figure 3.5: Stance to Flight commutative diagrams

positions are invariant. Thus the impact map for the coordinates can be expressed as

$$\Delta_{Stp}^q := \Pi_{Bsp}, \tag{3.40}$$

where Π_{Bsp} is as discussed in Section 3.2.3.

Next, the impact map for the velocities is derived as follows. Let τ_R to be the impulsive torque being applied at the pulley B_{spring} due to the stopper. Then the generalized external impulsive force acting on the system is obtained from the principle of virtual work as,

$$F_{\text{ext}} = \frac{\partial q_{Bsp_{st}}}{\partial q_e}{}^T \tau_R. \tag{3.41}$$

We have two constraints that need to be satisfied. The first condition is the trivial post impact velocity of the spring on the stance leg to be zero. The second condition is obtained by integrating the unconstrained dynamics, (3.7), over the duration of the instantaneous event. These conditions then are,

$$\dot{q}_{Bsp_{st}}^+ = 0 \implies \frac{\partial q_{Bsp_{st}}}{\partial q_e} \dot{q}_e^+ = 0. \tag{3.42}$$

$$D_e(q_e^+) \dot{q}_e^+ - D_e(q_e^-) = F_{\text{ext}}, \quad (3.43)$$

From (3.43), (3.42), assembling the constraints and solving for the post-impact velocity, we obtain the map,

$$\begin{aligned} \left(\Delta_{Stp}^{\dot{q}} \right)_{q_e^-} (\dot{q}_e^-) &= \begin{bmatrix} I & 0 \end{bmatrix} \\ &\begin{bmatrix} \frac{\partial q_{\text{BSPst}}}{\partial q_e} & 0 \\ D_e \left(\Delta_{Stp}^q(q_e^-) \right) & -\frac{\partial q_{\text{BSPst}}}{\partial q_e}^T \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ D_e(q_e^-) \dot{q}_e^- \end{bmatrix}, \end{aligned} \quad (3.44)$$

where Δ_{Stp}^q is as defined in (3.40). With this, the base and fiber components of the stance to flight transition map, (3.36), are completely defined.

3.2.7 Flight to Stance Transition Map

The robot physically transitions from flight phase to stance phase when the swing toe contacts the ground surface. It is assumed that there is no rebound or slipping when this contact occurs. Thus, mathematically, this transition occurs when the solution of (3.33) intersects the co-dimension one switching manifold defined as,

$$\mathcal{S}_{f \rightarrow s} := \{x_f \in TQ_f \mid H_{f \rightarrow s}(x_f) = 0\}, \quad (3.45)$$

where the threshold function $H_{f \rightarrow s} : TQ_{f \rightarrow s} \rightarrow \mathbb{R}$ is defined as $H_{f \rightarrow s}(x_f) = p_{\text{toesw}}^v$, with p_{toesw}^v being the vertical component of the swing toe.

The flight to stance transition map, $\Delta_{f \rightarrow s} : \mathcal{S}_{f \rightarrow s} \rightarrow TQ_s$, is defined as

$$\Delta_{f \rightarrow s}(x_f^-) := \begin{bmatrix} \Delta_{f \rightarrow s}^q(q_f^-) \\ (\Delta_{f \rightarrow s}^{\dot{q}})_{q_f^-}(\dot{q}_f^-) \end{bmatrix}, \quad (3.46)$$

where, $x_f^- = (q_f^-; \dot{q}_f^-) \in TQ_f$ is the final state of the flight phase. The base and fiber components, $\Delta_{f \rightarrow s}^q : Q_f \rightarrow Q_s$, $(\Delta_{f \rightarrow s}^{\dot{q}})_{q_f} : T_{q_f}Q_f \rightarrow T_{\Delta_{f \rightarrow s}^q(q_f)}Q_s$ define the transition maps for the configuration variables and their velocities, respectively. The initial state of the

$$\begin{array}{ccc}
\dot{q}_f^- \in T_{q_f^-} Q_f & \xrightarrow{(\Delta_{f \rightarrow s}^{\dot{q}})_{q_f^-}} & \dot{q}_s^+ \in T_{\Delta_{f \rightarrow s}^q(q_f^-)} Q_s \\
(D\Upsilon_f)_{q_f^-} \downarrow & \text{(fiber)} & \uparrow (D\Pi_s)_{\Delta_{Gnd}^q \circ \Upsilon_f(q_f^-)} \circ R \\
\dot{q}_e^- \in T_{\Upsilon_f(q_f^-)} Q_e & \xrightarrow{(\Delta_{Gnd}^{\dot{q}})_{\Upsilon_f(q_f^-)}} & \dot{q}_e^+ \in T_{\Delta_{Gnd}^q \circ \Upsilon_f(q_f^-)} Q_e
\end{array}
\tag{a}$$

$$\begin{array}{ccc}
q_f^- \in Q_f & \xrightarrow{\Delta_{f \rightarrow s}^q} & q_s^+ \in Q_s \\
\Upsilon_f \downarrow & \text{(base)} & \uparrow \Pi_s \circ R \\
q_e^- \in Q_e & \xrightarrow{\Delta_{Gnd}^q = id_{Q_e}} & q_e^+ \in Q_e
\end{array}
\tag{b}$$

Figure 3.6: Flight to Stance commutative diagrams

stance phase, $x_s^+ \in TQ_s$, is the post impact state and is obtained as,

$$x_s^+ = \Delta_{f \rightarrow s}(x_f^-). \tag{3.47}$$

The impact being modeled here is that of the swing leg impacting the ground. Mathematically, this is captured by the impact map $\Delta_{Gnd} : TQ_e \rightarrow TQ_e$ representing the impact with the ground. The base and fiber components of the flight to stance transition map can then be expressed using the impact map as,

$$\Delta_{f \rightarrow s}^q = \Pi_s \circ R \circ \Delta_{Gnd}^q \circ \Upsilon_f, \tag{3.48}$$

$$(\Delta_{f \rightarrow s}^{\dot{q}})_{q_f^-} = (D\Pi_s)_{\Delta_{Gnd}^q \circ \Upsilon_f(q_f^-)} \circ R \circ (\Delta_{Gnd}^{\dot{q}})_{\Upsilon_f(q_f^-)} \circ (D\Upsilon_f)_{q_f^-}, \tag{3.49}$$

such that diagram of Figure 3.6 commutes. Υ_f , Π_s are as in (3.27), (3.9) respectively, and R is the coordinate relabeling operator introduced earlier.

The rest of this section will focus on deriving the base and fiber components of the impact map Δ_{Gnd} .

As per earlier discussions, the impact with the ground does not result in a change in

the linkage positions. Further since both legs are off the ground, the springs are at their rest positions and thus the position of the transmission variables are invariant under this impact. Thus the impact map for the coordinates can be expressed as

$$\Delta_{Gnd}^q := \text{id}_{Q_e}. \quad (3.50)$$

Next, the impact map for the velocities is derived as follows. Let I_R to be the impulsive force on the foot due to the ground-foot impact. Furthermore let τ_R be the constraint force at the pulley B_{spring} to maintain the spring at its rest position (This is not an impact torque, but just a torque required to continue to enforce the constraint). Then the generalized external impulsive force acting on the system is obtained from the principle of virtual work as,

$$F_{\text{ext}} = \frac{\partial p_{\text{toesw}}}{\partial q_e}{}^T I_R + \left(\frac{\partial q_{B_{\text{spst}}}}{\partial q_e} \right)^T \tau_R. \quad (3.51)$$

We have three constraints that need to be satisfied at impact. The first condition is for the new swing leg to have zero spring velocity. The second condition is for the new stance toe to have zero velocity. The third constraint is obtained by integrating the unconstrained dynamics, (3.7), over the duration of the instantaneous event. These conditions then are,

$$\dot{q}_{B_{\text{spst}}}^+ = 0 \implies \frac{\partial q_{B_{\text{spst}}}}{\partial q_e} \dot{q}_e^+ = 0. \quad (3.52)$$

$$\dot{p}_{\text{toesw}} = 0 \implies \frac{\partial p_{\text{toesw}}}{\partial q_e} \dot{q}_e^+ = 0, \quad (3.53)$$

$$D_e(q_e^+) \dot{q}_e^+ - D_e(q_e^-) \dot{q}_e^- = F_{\text{ext}}, \quad (3.54)$$

From (3.53)-(3.54), assembling the constraints and solving for the post-impact velocity, we obtain the map,

$$\left(\Delta_{Gnd}^{\dot{q}} \right)_{q_e^-} (\dot{q}_e^-) = \begin{bmatrix} I & 0 & 0 \\ \frac{\partial q_{B_{\text{spst}}}}{\partial q_e} & 0 & 0 \\ \frac{\partial p_{\text{toesw}}}{\partial q_e} & 0 & 0 \\ D_e(\Delta_{Gnd}^q(q_e^-)) & -\frac{\partial p_{\text{toesw}}}{\partial q_e}{}^T & -\frac{\partial p_{B_{\text{spst}}}}{\partial q_e}{}^T \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ D_e(q_e^-) \dot{q}_e^- \end{bmatrix}, \quad (3.55)$$

where Δ_{Gnd}^q is as defined in (3.50). With this, the base and fiber components of the flight to stance transition map, (3.46), are completely defined.

3.2.8 Hybrid Model of Running

The hybrid model of running is based on the dynamics developed in Sections 3.2.2, 3.2.5, and transition maps derived in Sections 3.2.6, 3.2.7. The continuous dynamics with discrete state transitions between the stance and flight phases is represented as,

$$\Sigma_s : \begin{cases} \mathcal{X}_s = TQ_s \\ \mathcal{S}_{s \rightarrow f} = \{(x_s, u) \in \mathcal{X}_s \times \mathcal{U} \mid H_{s \rightarrow f}(x_s, u) = 0\} \\ \dot{x}_s = f_s(x_s) + g_s(x_s)u, & (x_s^-, u^-) \notin \mathcal{S}_{s \rightarrow f} \\ x_f^+ = \Delta_{s \rightarrow f}(x_s^-, u^-), & (x_s^-, u^-) \in \mathcal{S}_{s \rightarrow f} \end{cases} \quad (3.56)$$

$$\Sigma_f : \begin{cases} \mathcal{X}_f = TQ_f \\ \mathcal{S}_{f \rightarrow s} = \{x_f \in \mathcal{X}_f \mid H_{f \rightarrow s}(x_f) = 0\} \\ \dot{x}_f = f_f(x_f) + g_f(x_f)u, & x_f^- \notin \mathcal{S}_{f \rightarrow s} \\ x_s^+ = \Delta_{f \rightarrow s}(x_f^-), & x_f^- \in \mathcal{S}_{f \rightarrow s} \end{cases}$$

CHAPTER IV

Control Design for Walking

This chapter presents a feedback controller for achieving asymptotically stable, periodic walking gaits on MABEL. In addition to orbital stability, a key objective is to take advantage of the spring in the robot’s drivetrain that is placed in series with the leg-shape motor and q_{LS} . Inspired by analysis in [92, p. 1784] and [90, Chap. 6] for monopedal hoppers with compliance, this will be accomplished by controlling variables on the motor end of the spring and letting the joint end of the spring, which sees the large ground reaction forces, remain passive. In this way, the robot in closed-loop with the controller responds to impulsive forces at impact in a manner similar to a pogo stick. In particular, the closed-loop system will use the compliance to do negative work at impact (i.e., decelerating the center of mass and redirecting it upward¹), instead of it being done by the actuators, thereby improving the energy efficiency of walking.

It will be shown that the method of virtual constraints and hybrid zero dynamics [130] is flexible enough to accomplish the control objectives outlined above. The method of Poincaré [82] is used to verify stability of the closed-loop system. Prior to experimentally testing the controller, simulations with various model perturbations are performed to establish robustness of the designed controller. The controller is then experimentally validated on MABEL.

The rest of the chapter is as follows: Section 4.1 presents the virtual constraint design for walking, Section 4.2 presents the compliant zero dynamics for MABEL, Section

¹The double support phase of human walking is spring-like in the vertical direction and redirects the center of mass [97, 38]. The COM redirection is obtained here without a double support phase.

4.3 mathematically formalizes the event transitions between the subphases of the virtual constraints, Section 4.4 presents two cost functions for optimization and, finally, Section 4.5 presents the fixed points obtained by optimization.

4.1 Virtual Constraint Design for Walking

Recall that virtual constraints are holonomic constraints on the robot’s configuration variables that are asymptotically imposed through feedback control. They are used to synchronize the evolution of the robot’s links throughout a stride in order to synthesize a gait [130]. One virtual constraint is designed per independent actuator.

The virtual constraints are parametrized by θ , a strictly monotonic function of the joint configuration variables, and can be expressed in the form

$$y = h(q_s) = H_0 q_s - h_d(\theta), \quad (4.1)$$

where $H_0 q_s$ are the controlled variables, the variables on which the constraints are imposed, and $h_d(\theta)$ is the desired evolution of the each of the controlled variable.

If a feedback can be found such that y is driven asymptotically to zero, then $H_0 q_s \rightarrow h_d(\theta)$ and thus the controlled variables $H_0 q_s$ evolve according to the constraint $H_0 q_s = h_d(\theta)$. Here, the controlled variables are selected to be the rotor angle of the stance leg-shape motor, $q_{mLS_{st}}$, the swing leg variables, $q_{LA_{sw}}, q_{mLS_{sw}}$, and the absolute torso angle q_{Tor} . From hereon, the rotor angle of the stance leg-shape motor is simply referred to as stance motor leg shape.

4.1.1 Deciding What to Control

The torso is selected as a controlled variable instead of the stance leg angle, because, for MABEL, the torso represents over 65% of the mass of the robot, and hence the position of the torso heavily influences the gait. The stance motor leg shape, $q_{mLS_{st}}$, is chosen instead of the stance leg shape, $q_{LS_{st}}$, so that the joint side of the spring remains passive, as discussed above. Mathematically, with this choice, the spring variable will become a part of the zero dynamics, thereby rendering the zero dynamics compliant. From (3.1), if $q_{mLS_{st}}$ is held

constant, then $q_{\text{LS}_{\text{st}}}$ responds to the spring torque through $q_{\text{Bsp}_{\text{st}}}$. On the other hand, if $q_{\text{LS}_{\text{st}}}$ were selected as a controlled variable, then the actuator is forced to cancel the spring dynamics.

The swing leg virtual constraints are similar to the controlled variables on RABBIT, a robot without compliance. This is because under the assumption that the swing spring is at its rest position throughout stance, $q_{\text{Bsp}_{\text{sw}}} \equiv 0$, which from (3.1) shows that the motor leg shape, $q_{\text{mLS}_{\text{sw}}}$, is related to the leg shape, $q_{\text{LS}_{\text{sw}}}$, through a gear ratio.

In summary, the controlled variables are

$$H_0 q_s = \begin{bmatrix} q_{\text{mLS}_{\text{st}}} \\ q_{\text{LA}_{\text{sw}}} \\ q_{\text{mLS}_{\text{sw}}} \\ q_{\text{Tor}} \end{bmatrix}, \quad (4.2)$$

The desired evolution of each of the controlled variables are denoted by $h_{\text{mLS}_{\text{st}}}^d$, $h_{\text{LA}_{\text{sw}}}^d$, $h_{\text{mLS}_{\text{sw}}}^d$, and h_{Tor}^d respectively, and assembled as

$$h_d(\theta) = \begin{bmatrix} h_{\text{mLS}_{\text{st}}}^d(\theta) \\ h_{\text{LA}_{\text{sw}}}^d(\theta) \\ h_{\text{mLS}_{\text{sw}}}^d(\theta) \\ h_{\text{Tor}}^d(\theta) \end{bmatrix}. \quad (4.3)$$

For MABEL, we choose θ to be the absolute angle formed by the virtual compliant leg relative to the ground. This is indicated in Figure 4.1 and defined as

$$\theta(q_s) = \pi - q_{\text{LA}_{\text{st}}} - q_{\text{Tor}}. \quad (4.4)$$

4.1.2 Specification of the Constraints

Virtual constraints for the stance phase of MABEL are inspired by the constraints designed for Thumper in [90, 91]. The stance phase is broken up into subphases: the motor-

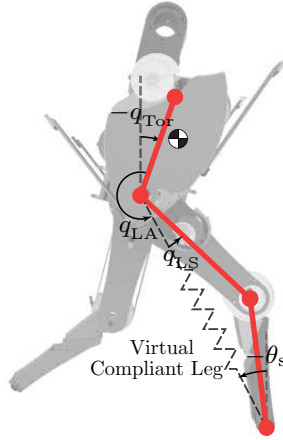


Figure 4.1: The coordinate system used for the linkage is indicated. Angles are positive in the counter clockwise direction. The *virtual compliant leg* is created by the drivetrain through a set of differentials.

compression phase (*mc*), the stance-compression phase (*sc*), the stance-injection phase (*si*), and the stance-decompression phase (*sd*). The details of these subphases are given later in the section. Figure 4.2 illustrates the evolution of each of these constraints on $q_{mLS_{st}}$, $q_{LA_{sw}}$, $q_{mLS_{sw}}$, and q_{Tor} .

The reason behind breaking up the stance phases into four subphases is to facilitate the design of virtual constraints that effectively make use of the compliance. A key idea is to hold the stance motor leg shape at a constant value shortly after impact in order to allow the spring (which is in series with this actuator) to absorb the impact shock entirely. Note that if the motor position is held constant, then its velocity is zero and the motor performs no mechanical work. The spring then does the negative work of decelerating the center of mass and redirecting it upwards; in other words, the spring stores the impact energy and returns it later to the gait instead of the actuator doing negative work and dissipating it as heat. This effectively preserves the natural compliant dynamics of the system and prevents the actuator from fighting the spring.

Another key subphase involves the torso. Because it is heavy, we have observed that making the pre-impact torso velocity close to zero at the end of the gait helps in avoiding excessive forward pitching of the torso just after swing leg impact. This is achieved by designing the torso virtual constraint such that, before impact, the torso position is constant and its velocity is zero; see phase *sd* in Figure 4.2.

Remark 4.1. The choice of the variables to be controlled in the virtual constraints makes the zero dynamics compliant. The choice of the evolution of the virtual constraints facilitates efficient use of the compliance.

In our design of the virtual constraints for MABEL, we use the framework for virtual constraints with subphases developed in Appendix A, with the index set for the subphases given by

$$\mathcal{P} := \{mc, sc, si, sd\}, \quad (4.5)$$

and with the index set for the virtual constraints given by

$$\mathcal{V} := \{\text{mLS}_{\text{st}}, \text{LA}_{\text{sw}}, \text{mLS}_{\text{sw}}, \text{Tor}\}. \quad (4.6)$$

Further, we choose $M = 5$ in (A.2), and impose \mathcal{C}^1 continuity between successive subphases. This ensures continuity of position and velocity at the boundary of two phases of a virtual constraint. However, acceleration, and consequently, the actuator torques, are allowed to be discontinuous at phase boundaries.

4.1.3 Stance Motor Leg-shape Virtual Constraint

The desired evolution of the stance motor leg-shape position, $h_{\text{mLS}_{\text{st}}}^d$, is as follows. During the motor-compression phase, the velocity of the motor leg-shape immediately after impact, $\dot{q}_{\text{mLS}_{\text{st}}}^+$, is usually nonzero and is smoothly brought to zero by the end of the motor-compression phase, i.e., $\dot{q}_{\text{mLS}_{\text{st}}}^{mc-} = 0$.

Throughout the stance-compression phase, the leg-shape motor position is kept at a constant angle $q_{\text{mLS}_{\text{st}}}^{sc}$. With the motor position locked, the bending of the stance knee compresses the spring. The phase lasts until $q_{\text{BSP}_{\text{st}}} = q_{\text{BSP}_{\text{st}}}^{sc+}$ with $\dot{q}_{\text{BSP}} < 0$, the point at which the spring decompresses to a value near the nominal spring compression at mid-stance, a typical value being five degrees. This ensures that the impact kinetic energy is first stored in the spring and then returned to the gait without the actuator performing unnecessary negative work on the leg shape, $q_{\text{LS}_{\text{st}}}$.

The stance-injection phase starts with the spring just decompressed to $q_{\text{BSP}_{\text{st}}}^{sc+}$. The ac-

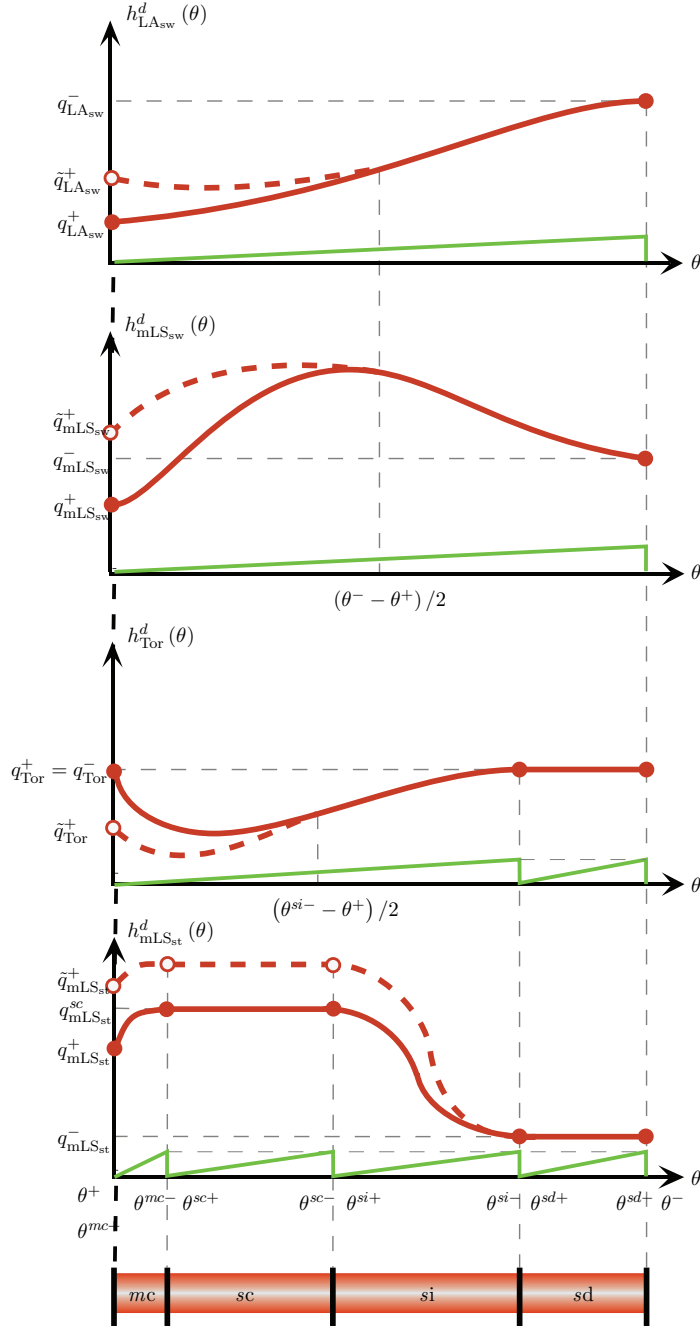


Figure 4.2: The general shape of the stance phase virtual constraints. The thick solid lines illustrate the evolution of the virtual constraints as a function of θ . Each virtual constraint is broken into subphases (mc , sc , si , sd) and each subphase is locally expressed by a 5th order Bézier polynomial. The thin lines show the evolution of corresponding local s that parametrizes the local Bézier curve and goes from 0 to 1. The subphases can be combined as, for instance, in the torso virtual constraint, which lumps the first three subphases together and normalizes s appropriately. The swing leg virtual constraints combine all four phases. The thick dashed lines are correction polynomials introduced to create hybrid invariance of the zero dynamics and are discussed in Section 4.6.

tuator then rapidly repositions the motor shaft to a new desired position, $q_{\text{mLS}_{\text{st}}}^-$. Under nominal conditions, this straightens the leg during mid-stance for ground clearance of the swing leg. Under large perturbations, this motion will cause the actuator to inject (or remove) energy through compression (or decompression) of the spring by rapidly repositioning the motor end of the compliance.

Following the stance-injection phase, the motor shaft is maintained at the position $q_{\text{mLS}_{\text{st}}}^-$ throughout the stance-decompression phase, waiting for the spring to decompress again in preparation for leg touchdown.

The virtual constraints for the stance motor leg shape are depicted in Figure 4.2. The thick solid line is the virtual constraint, and the thin line is the local s that parametrizes the local Bézier polynomial. The figure also shows the virtual constraints for the other controlled variables. Appendix A provides further details regarding choosing the the Bézier polynomial coefficients for each subphase.

4.1.4 Torso Virtual Constraint

The desired evolution of the torso angle, h_{Tor}^d , does not need to be as finely specified; it's evolution will be primarily left to optimization, which will be discussed in Section 4.4. The motor-compression, stance-compression, and stance-injection phases, are combined into a single phase. This phase serves as a transient phase that drives the torso in a smooth manner from the initial configuration, $(q_{\text{Tor}}^+, \dot{q}_{\text{Tor}}^+)$, to the final configuration, $(q_{\text{Tor}}^-, \dot{q}_{\text{Tor}}^- = 0)$, in preparation for impact.

During the stance-decompression phase, the torso is held constant in preparation for impact. Simulations with the model and experiments with a simpler PD controller detailed in [49] showed that achieving a nearly zero pre-impact velocity tends to prevent the heavy torso from excessively pitching after impact.

4.1.5 Swing Leg Virtual Constraints

The desired evolution of the swing leg angle, $h_{\text{LA}_{\text{sw}}}^d$, and motor leg shape position, $h_{\text{mLS}_{\text{sw}}}^d$ are the simplest of all of the constraints as no subphases are used. A single virtual constraint on swing leg angle is designed to bring the leg forward, preparing it for impact

with a desired step length. The constraint on swing motor leg shape is responsible for lifting the swing leg from the ground, avoiding foot scuffing during the gait, and extending the leg before impact. These two constraints are similar to RABBIT [132].

4.1.6 Discussion

The use of subphases in the evolution of the stance motor leg shape and torso introduces additional independent parameters to be specified in the constraint design. One benefit is that it approximately decouples the evolution of these angles from one phase to another; changing the evolution in one phase does not strongly affect the other as long as the boundary condition is maintained. This facilitates intuitively specifying the initial shape of the virtual constraints and makes the optimization task easier. For a list of independent parameters to be found by optimization, refer to Table B.1 in Appendix B.

For later use, we can organize the virtual constraints for each phase separately. For each $p \in \mathcal{P}$, we can define the output,

$$y_p = h_p(q_s, \alpha_p) = H_0^p(q_s) - h_d^p(\theta, \alpha_p), \quad (4.7)$$

and,

$$h_d^p(\theta, \alpha_p) = \begin{bmatrix} h_{\text{mLS}_{\text{st}}}^{d,p}(\theta) \\ h_{\text{LA}_{\text{sw}}}^{d,p}(\theta) \\ h_{\text{mLS}_{\text{sw}}}^{d,p}(\theta) \\ h_{\text{Tor}}^{d,p}(\theta) \end{bmatrix}. \quad (4.8)$$

The Bézier coefficients for each phase can be organized as,

$$\alpha_p = \begin{bmatrix} \alpha_{\text{mLS}_{\text{st}}}^p \\ \alpha_{\text{LA}_{\text{sw}}}^p \\ \alpha_{\text{mLS}_{\text{sw}}}^p \\ \alpha_{\text{Tor}}^p \end{bmatrix}. \quad (4.9)$$

Remark 4.2. Both the local virtual constraint h_d^p and the local selection of the controlled variables H_0^p can be modified for each subphase resulting in (4.7). Here we only change the

parameters used in the Bézier polynomials α_v^p and leave $H_0^p = H_0$ as defined in (4.2), and $h_d^p = h_d$ as defined in (4.3), for each phase $p \in \mathcal{P}$.

4.2 Zero Dynamics for Walking

The organization of the stance phase into four subphases creates four continuous dynamics and discrete transitions between them. As discussed in Section 4.1.6, for each phase $p \in \mathcal{P}$, an output function y_p has been associated with the continuous stance dynamics defined in (3.14). The zero dynamics is defined as the maximal internal dynamics of the system that is compatible with the output being identically zero [64]. Differentiating the output twice with respect to time results in

$$\frac{d^2 y_p}{dt^2} = L_{f_s}^2 h_p(x_s, \alpha_p) + L_{g_s} L_{f_s} h_p(q_s, \alpha_p) u, \quad (4.10)$$

where $L_{g_s} L_{f_s} h_p(q_s, \alpha_p)$, the decoupling matrix, has full rank. Under the conditions of [130, Lemma 5.1],

$$u^*(x_s, \alpha_p) := -(L_{g_s} L_{f_s} h_p(q_s, \alpha_p))^{-1} L_{f_s}^2 h_p(x_s, \alpha_p), \quad (4.11)$$

is the unique control input that renders the smooth four-dimensional embedded submanifold

$$\mathcal{Z}_{\alpha_p} = \{x_s \in TQ_s \mid h_p(q_s, \alpha_p) = 0, L_{f_s} h_p(x_s, \alpha_p) = 0\} \quad (4.12)$$

invariant under the stance dynamics (3.14); that is, for every $z \in \mathcal{Z}_{\alpha_p}$,

$$f_p^*(z) := f_s(z) + g_s(z) u^* \in T_z \mathcal{Z}_{\alpha_p}. \quad (4.13)$$

Achieving the virtual constraints by zeroing the corresponding outputs reduces the dimension of the system by restricting its dynamics to the submanifold \mathcal{Z}_{α_p} embedded in the continuous-time state space TQ_s . \mathcal{Z}_{α_p} is called the zero dynamics manifold and the restriction dynamics $\dot{z} = f_p^*|_{\mathcal{Z}_{\alpha_p}}(z)$ is called the zero dynamics.

From Lagrangian dynamics (the derivation is standard [130, Chap. 5] and skipped for

sake of brevity), a valid set of coordinates on \mathcal{Z}_{α_p} is

$$x_{zd}^p = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \end{bmatrix} = \begin{bmatrix} \theta \\ q_{\text{BSPst}} \\ \frac{\partial \mathcal{L}_s}{\partial \dot{q}_{\text{BSPst}}} \\ \frac{\partial \mathcal{L}_s}{\partial \dot{q}_{\text{Tor}}} \end{bmatrix}. \quad (4.14)$$

This set of coordinates explicitly contains the B_{spring} variable, which illustrates clearly that the zero dynamics is compliant:

$$\dot{x}_{zd}^p = \begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \\ \dot{\xi}_4 \end{bmatrix} = \begin{bmatrix} L_{f_s} \theta \\ L_{f_s} q_{\text{BSPst}} \\ \frac{\partial \mathcal{L}_s}{\partial q_{\text{BSPst}}} + \tau_{sp} \\ \frac{\partial \mathcal{L}_s}{\partial q_{\text{Tor}}} \end{bmatrix}. \quad (4.15)$$

4.3 Event Transitions

The division of the stance phase into subphases when specifying the virtual constraints in Section 4.1 necessitates the specification of the transition maps between the subphases. In preparation for the next section, we model the hybrid dynamics on the zero dynamics manifold by concatenating the solutions of the parameter-dependent hybrid systems for each subphase

$$\Sigma_p : \begin{cases} x_p \in \mathcal{Z}_{\alpha_p} \\ \mathcal{S}_{p \rightarrow q} = \{x_p \in \mathcal{Z}_{\alpha_p} \mid H_{p \rightarrow q}(x_p) = 0\} \\ \dot{x}_p = f_p^*(x_p), & x_p^- \notin \mathcal{S}_{p \rightarrow q} \\ x_q^+ = \Delta_{p \rightarrow q}(x_p^-), & x_p^- \in \mathcal{S}_{p \rightarrow q}. \end{cases}$$

The model captures the continuous-time dynamics of the system in phase $p \in \mathcal{P}$ and the discrete transition to phase $q \in \mathcal{P}$, with the only valid choice of transitions for walking being $(p, q) \in \{(mc, sc), (sc, si), (si, sd), (sd, mc)\}$.

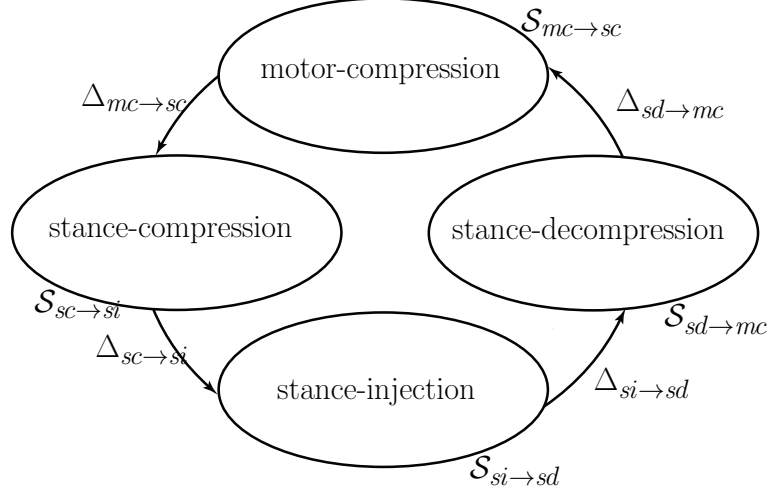


Figure 4.3: The hybrid system for walking with continuous-time phases and discrete event transitions.

The switching surfaces, $\mathcal{S}_{p \rightarrow q}$, for the transitions for walking are defined by the zero level sets of the corresponding threshold functions $H_{p \rightarrow q} : TQ_s \rightarrow \mathbb{R}$, which are given below,

$$\begin{aligned}
 H_{mc \rightarrow sc} &:= \theta - \theta^{mc-} \\
 H_{sc \rightarrow si} &:= q_{\text{Bsp}_{\text{st}}} - 5^\circ \\
 H_{si \rightarrow sd} &:= \theta - \theta^{si-} \\
 H_{sd \rightarrow mc} &:= p_{\text{toe}_{\text{sw}}}^v.
 \end{aligned} \tag{4.16}$$

The transition maps, $\Delta_{p \rightarrow q} : \mathcal{S}_{p \rightarrow q} \rightarrow TQ_s$, provide the initial conditions for the ensuing phase $q \in \mathcal{P}$, and are given below,

$$\begin{aligned}
 \Delta_{mc \rightarrow sc} &:= id \\
 \Delta_{sc \rightarrow si} &:= id \\
 \Delta_{si \rightarrow sd} &:= id \\
 \Delta_{sd \rightarrow mc} &:= \Delta_{s \rightarrow s},
 \end{aligned} \tag{4.17}$$

where id is the identity map and $\Delta_{s \rightarrow s}$ is defined in (3.16).

The event transitions are indicated in Figure 4.3. To find a set of values for the independent parameters of the constraint design specified in Section 4.1, we employ the above

hybrid system and formulate the problem as a constrained optimization.

4.4 Gait Design Through Optimization

A periodic walking gait is designed by selecting the free parameters in the virtual constraints. As in [134, 130], this is most easily done by posing an optimization problem, such as minimum energy per step length, subject to constraints to meet periodicity, workspace and actuator limitations, and desired walking speed. The equations of the compliant zero dynamics, which are of reduced dimension compared to the full dynamics, are employed in the optimization for efficiency of computation.

The nonlinear constrained optimization routine `fmincon` of MATLAB's Optimization Toolbox is used to perform the numerical search for desired gaits. The quantities involved in optimization are the scalar cost function to be minimized, J , the vector of equality constraints, EQ, and the vector of inequality constraints, INEQ. The optimization algorithm, equality and inequality conditions are given in Appendix B and the list of optimization parameters is specified in Table B.1.

Several popular cost functions for bipedal gait design are given in [130, Sec. 6.3.3]. Here, two cost criteria are used in the optimization process. First we use a nominal cost function, as used in RABBIT in [132], consisting of the integral over a step of squared torque divided by distance traveled,

$$J_{nom}(\alpha) = \frac{1}{p_{\text{toe}_{\text{sw}}}^h(q_s^-)} \int_0^{T_I} \|u(t)\|^2 dt, \quad (4.18)$$

where T_I is the step duration and $p_{\text{toe}_{\text{sw}}}^h$ is the step length. Minimizing this cost function tends to reduce peak torque demands and minimizes the electrical energy consumed per step.

Next we use a cost function that quantifies the mechanical energy consumed. The specific mechanical cost of transport, c_{mt} is introduced in [26, 27] as means of quantifying energy consumed for bipedal locomotion. c_{mt} is the energy consumed per unit weight per

unit distance traveled and can be defined as a cost function,

$$J_{cmt} = \frac{\int_0^{T_I} \sum_{i=1}^4 E_i(t) dt}{Mgd}, \quad (4.19)$$

where M is the mass of the robot, g is the acceleration due to gravity, d is the distance traveled, and

$$E_i(t) = \begin{cases} u_i(t) \dot{q}_i(t) & u_i(t) \dot{q}_i(t) > 0 \\ 0 & u_i(t) \dot{q}_i(t) \leq 0 \end{cases} \quad (4.20)$$

Mechanical power can be either positive (energy is injected) or negative (energy is absorbed). Some authors, [51], consider the absolute mechanical power while defining the c_{mt} , whereas the definition in (4.19), (4.20) does not take into account any negative work that is performed by the actuators, the idea being that if the actuators were redesigned, energy could be absorbed mechanically through a friction brake or electronically through regenerative braking.

4.5 Fixed Point for Walking

This section presents a nominal fixed point of 0.8 m/s obtained by applying the optimization procedure outlined in Section 4.4 to the virtual constraints of Section 4.1, and with the cost function (4.18). Figure 4.4 illustrates the nominal evolution of the virtual constraints and other configuration variables for one step. It is seen that the stance motor leg shape is held constant for the first part of the gait right after impact, and both the stance motor leg shape and the torso are held constant towards the final part of the gait. Interestingly, the torso moves less than two degrees throughout the step.

Figure 4.5 illustrates the evolution of the leg shape and the stance B_{spring} variables. Notice that the spring compresses to its peak value, and the $sc \rightarrow si$ transition is triggered as the spring decompresses to five degrees. The injection of energy in the si -phase causes the spring to compress again. Figure 4.6 illustrates the actuator torques used to realize the gait. These torques are small in comparison to the peak torque capacities of the actuators: 30 Nm at u_{mLA} and 55 Nm at u_{mLS} . The torques are discontinuous at phase boundaries, as noted earlier, due to the choice of the virtual constraints being \mathcal{C}^1 at phase boundaries.

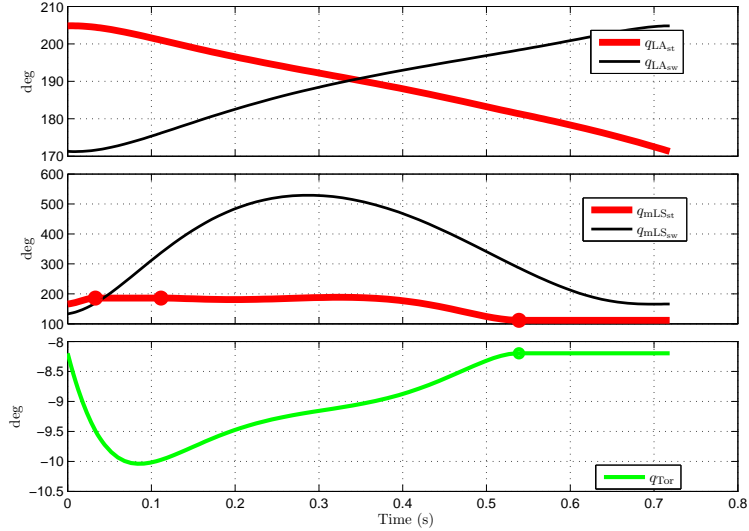


Figure 4.4: Evolution of the virtual constraints and configuration variables for a nominal fixed point (periodic walking gait) at a speed of 0.8 m/s and step length 0.575 m. The dots on the stance motor leg-shape virtual constraint illustrate the location of transition between consecutive subphases.

Figure 4.7 illustrates the evolution of the swing leg height and the vertical position of the center of mass (COM) of the robot. The COM moves downward immediately after impact, before reversing course and following a roughly parabolic path. Such a trajectory more closely resembles that of a human [73] than that of a robot with rigid links and rigid gearing. Figure 4.8 illustrates the vertical component of the ground reaction force at the stance leg. A double hump is observed and can be related to what is predicted for walking with compliant legs in [38]. The specific cost of mechanical transport for this nominal gait is, $c_{mt} = 0.0452$. The corresponding power plot is very similar to the power plot for the next designed gait and is not shown.

A second walking gait was designed, this time using cost function (4.19), which optimizes for the specific cost of mechanical transport. The optimization terminated with a value of $c_{mt} = 0.0385$ which is over 10% lower than that for the nominal gait. For this fixed point, Figure 4.9 compares the total power provided by the stance leg shape motor to the total power at stance leg shape, where the latter is the sum of the actuator and compliance power. It is clear that the spring is doing the vast majority of the negative work that is necessary on the stance leg.

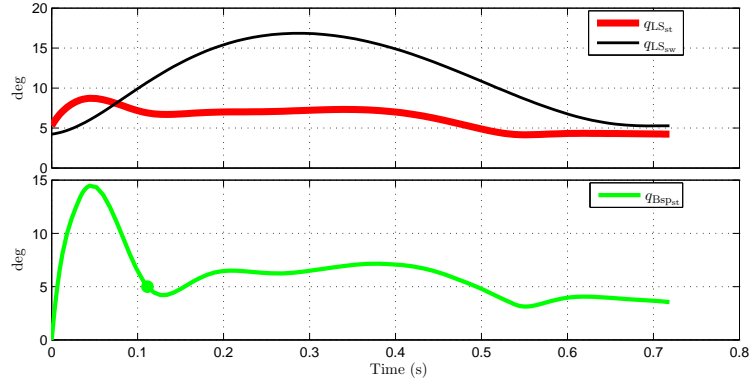


Figure 4.5: Evolution of the leg shape and stance B_{spring} variables corresponding to the nominal fixed point. The dot on the B_{spring} plot illustrates the location of the sc to si event transition and corresponds to $q_{Bsp_{st}} = 5^\circ$.

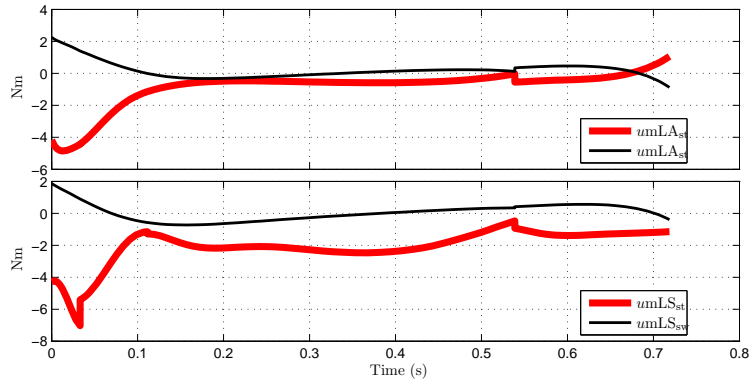


Figure 4.6: Actuator torques corresponding to the nominal fixed point. Note that the torques are discontinuous at subphase boundaries, due to the choice of C^1 continuity of the virtual constraints at subphase boundaries as per Appendix A.

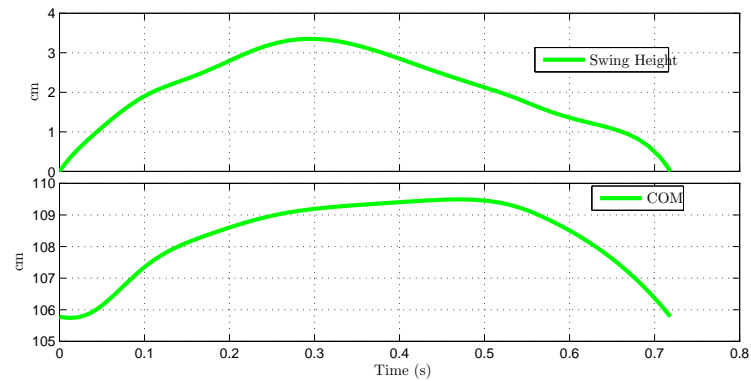


Figure 4.7: Evolution of swing leg height and vertical center of mass (COM) of the robot for the nominal fixed point. The COM trajectory more closely resembles that of a human gait than of a robot with rigid links since the COM moves downward immediately after impact, before reversing course and following a roughly parabolic path.

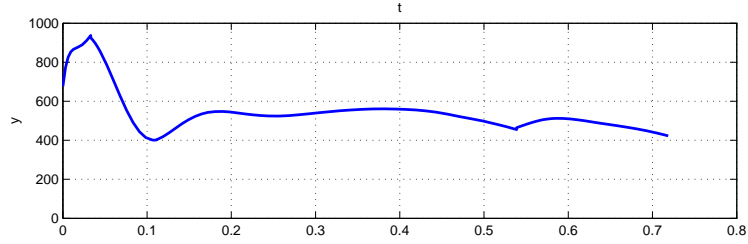


Figure 4.8: Ground reaction force for the nominal walking fixed point. A double hump is observed and can be related to what is predicted for walking with compliant legs in [38].

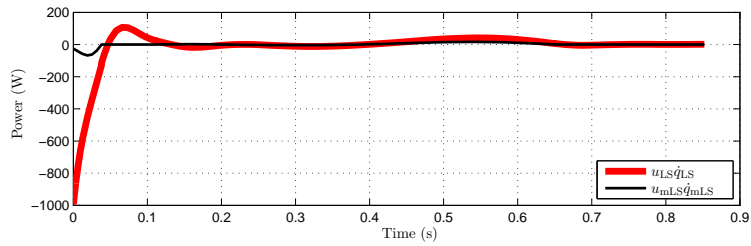


Figure 4.9: Power plot of a fixed point obtained by optimization of (4.19), specific cost of mechanical transport. The thick line illustrates the total instantaneous power at the leg shape from the actuator and the spring, and the thin line illustrates the instantaneous power at the leg shape from the motor alone. The difference is the energy that is saved and that would otherwise have to be provided by the actuator if the spring were absent. This plot shows the significant energy economy at impacts due to the presence of the compliance in the transmission.

4.6 Closed-loop Design and Stability Analysis

The feedback presented in (4.11) renders the zero dynamics manifold invariant under the stance phase dynamics. It is used in the optimization process of gait design in order to evaluate the torques along a solution of the model respecting the virtual constraints. The feedback (4.11) does not however render the solution stable or attractive in any way. In the following, two controllers based on the classic input-output linearizing controller

$$u = u^*(x_s, \alpha_p) - L_{g_s} L_{f_s} h_p(q_s, \alpha_p)^{-1} \left(\frac{K_{p,P}}{\epsilon^2} y + \frac{K_{p,D}}{\epsilon} \dot{y} \right), \quad (4.21)$$

where $p \in \{mc, sc, si, sd\}$, are discussed.

4.6.1 A PD + Feedforward Controller

With an eye toward experimental implementation, we look at successful controllers that have been employed to enforce virtual constraints in experiments. For RABBIT, it was possible to implement the virtual constraints through a simple PD controller [132], per

$$u = -K_P y - K_D \dot{y}$$

for y given by (4.1), and \dot{y} computed numerically. On MABEL, such a controller (employed in experiments in [49]) resulted in virtual constraints that were not accurately achieved due to large tracking errors, and attempts at reducing the errors with high controller gains were unsuccessful. See Section 5.2.6 for discussion of this point.

To address this, the vector of nominal control torques u^* from (4.11) is incorporated as a feedforward term in the PD controller. In particular, along the nominal orbit, for each of the actuated variables $m \in \mathcal{M}$ and for each of the phases $p \in \mathcal{P}$, u^* is regressed against θ with 5th order Bézier polynomials to obtain the Bézier coefficients β_p^m and resulting in the controller

$$u_{exp} = u^*(\theta, \beta) - K_P y - K_D \dot{y}, \quad (4.22)$$

where, y is as defined in (4.1), \dot{y} still computed numerically, and $\beta = (\beta_p^m)$.

The stability of the fixed-points with the proposed closed-loop controller (4.22) can be tested numerically using a Poincaré map $P : \mathcal{S} \rightarrow \mathcal{S}$ with the switching surface taken to be the switching surface at the $si \rightarrow sd$ event transition, i.e., $\mathcal{S} = \mathcal{S}_{si \rightarrow sd}$, and

$$P(x_s) = \phi(T_I \circ \Delta_{si \rightarrow sd}(x_s), \Delta_{si \rightarrow sd}(x_s)), \quad (4.23)$$

where, $\phi(t, x_0)$ denotes the maximal solution of (3.14), with initial condition x_0 at time $t_0 = 0$ and with u as defined in (4.22). Finally, T_I is the time-to-impact function defined in the usual way [130, p. 94].

Using the Poincaré return map (4.23), we can numerically calculate the eigenvalues of its linearization about the fixed-point. The analysis shows that the walking gait obtained by optimizing (4.18) and with the closed-loop controller (4.22) is exponentially stable with a dominant eigenvalue of 0.6921.

Similarly, the gait obtained from optimizing (4.19) is also exponentially stable, with a dominant eigenvalue of 0.8194.

4.6.2 Hybrid Invariance

The above controllers are not hybrid invariant. It was discovered in [83, 84] that, in the presence of compliance, while the feedback controller (4.21) will render the zero dynamics manifold of a given phase invariant under the continuous dynamics, it will not necessarily render it invariant under the transition maps, that is, at transitions from one phase to another, invariance is lost. The loss of invariance manifests itself as an impulsive disturbance to the control law at each transition off the periodic orbit. These perturbations do not prevent asymptotic stability from being achieved, but they do cause the actuators to do more work. The reference [84] proposed a supplemental event-based controller that eliminates this issue and, in fact, creates a *hybrid zero dynamics* for the closed-loop system, that is, the zero dynamics manifold is invariant under the continuous dynamics as well as the transition maps.

For the related robot, Thumper, [90, 91] propose an event-based control at each phase transition. This is not practical here, however, because we have certain phases with ex-

tremely small duration (the mc phase for instance). Instead, we create a hybrid zero dynamics by updating parameters only at the impact event (swing leg contacts the ground).

Following [84, 48], the virtual constraints are modified stride to stride so that they are compatible with the initial state of the robot at the beginning of each step. The new output for the feedback control design is,

$$\begin{aligned} y_c &= h(q_s, y^{s+}, \dot{y}^{s+}) \\ &= H_0 q_s - h_d(\theta) - h_c(\theta, y^{s+}, \dot{y}^{s+}). \end{aligned} \quad (4.24)$$

The output consists of the previous output, (4.1), and an additional correction term that depends on the previous output evaluated at the beginning of the step, specifically, $y^{s+} = H_0 q_s^+ - h_d(\theta^+)$, and $\dot{y}^{s+} = H_0 \dot{q}_s^+ - \frac{\partial h_d(\theta)}{\partial \theta} \dot{\theta}^+$. The values of y^{s+} , and \dot{y}^{s+} are determined at the beginning of each step and held constant throughout the step. The function h_c is taken here as

$$h_c(\theta) = \begin{bmatrix} 0 \\ h_{\text{LA}_{\text{sw}}}^c(\theta) \\ h_{\text{mLS}_{\text{sw}}}^c(\theta) \\ h_{\text{Tor}}^c(\theta) \end{bmatrix}, \quad (4.25)$$

with each $h_v^c(\theta)$, $v \in \mathcal{V} \setminus \{\text{mLS}_{\text{st}}\}$ taken to be twice continuously differentiable functions of θ such that,

$$\begin{cases} h_v^c(\theta, y^{s+}, \dot{y}^{s+}) &= y^{s+} \\ \frac{\partial h_v^c}{\partial \theta_s}(\theta^+) &= \frac{\dot{y}^{s+}}{\dot{\theta}_s^+} \\ h_v^c(\theta, y^{s+}, \dot{y}^{s+}) &= 0, \frac{\theta^+ + \theta_v^{mc+}}{2} \leq \theta \leq \theta_v^{mc-} \end{cases}. \quad (4.26)$$

With h_v^c designed this way, the initial errors of the output and its derivative are smoothly joined to the original virtual constraint at the middle of the first phase of the corresponding virtual constraint. By the choice of $\theta_{\text{LA}_{\text{sw}}}^{mc-}$, $\theta_{\text{mLS}_{\text{sw}}}^{mc-}$ and $\theta_{\text{Tor}}^{mc-}$ (defined in Section 4.1), the joining of the swing leg virtual constraints occurs at the middle of the step, while the joining for the torso virtual constraint occurs earlier, at the middle of the combined phases $mcscsi$. This is illustrated in Figure 4.2 with thick dashed lines.

As noted in the definition of h_c in (4.25), we have selected $h_{\text{mLS}_{\text{st}}}^c \equiv 0$ since the mc phase

is too short to handle significant transients without large actuator torques, and further we want to enforce the virtual constraint in the *sc* phase to be constant in order to effectively use the compliance. To overcome this, we propose an event-based control action specific for the mLS_{st} virtual constraint that updates $\alpha_{mLS_{st}}^{mc}, \alpha_{mLS_{st}}^{sc}, \alpha_{mLS_{st}}^{si}$ at the beginning of each step such that during the *mc* phase, the virtual constraint only drives the motor leg shape velocity to zero, and during the *sc* phase, the virtual constraint keeps the motor shaft locked at a constant position. Not until the *si* phase does the modified virtual constraint smoothly join the nominal virtual constraint. This correction term is also illustrated in Figure 4.2 with thick dashed lines.

Under the new control law defined by (4.24), the behavior of the robot is completely defined by the event transition maps and the swing phase zero dynamics, with h_d replaced by $h_d + h_c$. The stability of the fixed-point x^* can now be tested numerically using a restricted Poincaré map $\rho : \mathcal{S} \cap \mathcal{Z} \rightarrow \mathcal{S} \cap \mathcal{Z}$ where $\mathcal{Z} = \{x_s \in TQ_s \mid y_c(q_s) = 0, \dot{y}_c(q_s) = 0\}$, the switching surface is taken to be the switching surface at the *si* \rightarrow *sd* event transition, i.e., $\mathcal{S} = \mathcal{S}_{si \rightarrow sd}$, and

$$\rho(x_s) = \phi(T_I \circ \Delta_{si \rightarrow sd}(x_s), \Delta_{si \rightarrow sd}(x_s)), \quad (4.27)$$

where, $\phi(t, x_0)$ denotes the maximal solution of (3.14), with initial condition x_0 at time $t_0 = 0$ and u as defined in (4.21). Hybrid invariance is achieved because the transition map for these events, $\Delta_{si \rightarrow sd}$ is the identity map, and $\Delta_{si \rightarrow sd}(\mathcal{S}_{si \rightarrow sd} \cap \mathcal{Z}) \subset \mathcal{S}_{si \rightarrow sd} \cap \mathcal{Z}$.

Using the restricted Poincaré return map (4.27), we can numerically calculate the eigenvalues of its linearization about the fixed-point. For the gait obtained by optimizing (4.18), we obtain the eigenvalues

$$eig\left(\frac{\partial \rho(x_s)}{\partial x_s}\right) = \begin{bmatrix} 0.7258 \\ 2.6380e - 5 \\ -1.8001e - 6 \end{bmatrix}. \quad (4.28)$$

From [84, Cor. 2], the feedback (4.21) and (4.24) renders the periodic orbit of the closed-

loop system exponentially stable for ϵ in (4.21) sufficiently small, and K_P, K_D such that $\lambda^2 + K_D\lambda + K_P = 0$ is Hurwitz.

The orbit obtained from optimizing (4.19) is also exponentially stable, with a dominant eigenvalue 0.7065.

4.6.3 Robustness Study of PD Controller

In this section we study the robustness of the controllers proposed in the previous section. With an eye towards controller simplicity for experimental implementation, we only consider the simplest controller among the proposed controllers. This is the feedforward plus PD controller (4.22) developed in Section 4.6.1. We study the robustness of (4.22) to various perturbations in the model. First, a full state feedback is performed with the feedforward plus PD controller, providing a best case scenario for the controller. Next, the positions variables are quantized, corresponding to the physical quantization due to the finite resolution of the encoders, and the velocities are estimated from the quantized position using numerical differentiation. Following this, several model perturbations are introduced sequentially. These include perturbations such as errors in the torso mass, spring stiffness, torso center of mass position, and deviations in initial conditions. Figures 4.10, 4.11 illustrates the corresponding outputs and torques under the various perturbations respectively.

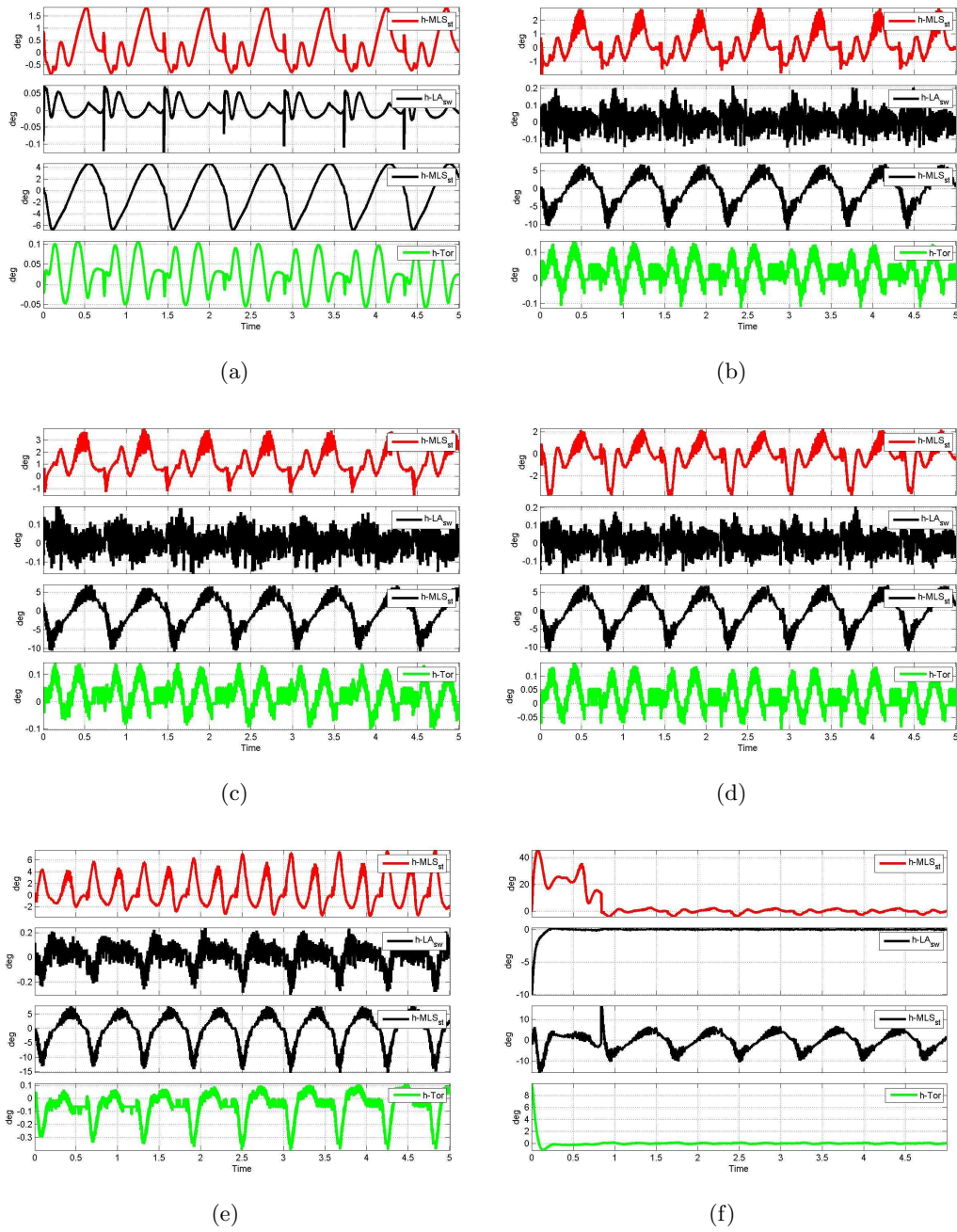


Figure 4.10: Outputs obtained by simulation of the system with the feedforward plus PD controller with (a) full state feedback, (b) q quantized, \dot{q} estimated, (c) torso mass increased by 5%, (d) spring constant increased by 10%, (e) torso COM rotated forward by 5° , and (f) initial condition changed for Torso to lean back by 10° . In (c) - (e), q was quantized, and \dot{q} estimated

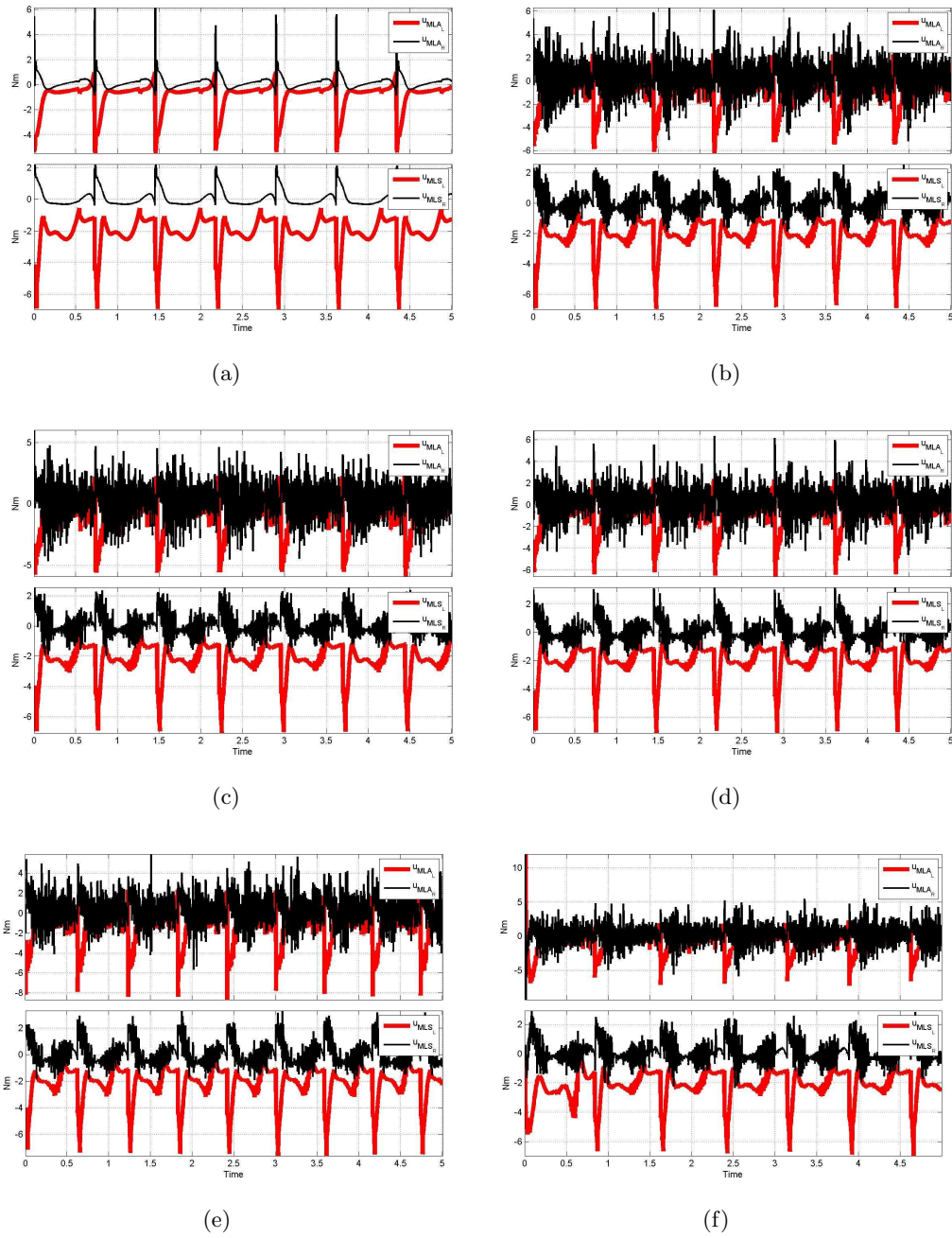


Figure 4.11: Torque obtained by simulation of the system with the feedforward plus PD controller with (a) full state feedback, (b) q quantized, \dot{q} estimated, (c) torso mass increased by 5%, (d) spring constant increased by 10%, (e) torso COM rotated forward by 5° , and (f) initial condition changed for Torso to lean back by 10° . In (c) - (e), q was quantized, and \dot{q} estimated

CHAPTER V

Walking Experiments

This chapter documents experimental implementations of the controllers of Section 4.6 in various walking scenarios. Figure 5.1 depicts the experimental setup. To illustrate the power and limitations of the proposed methods, five experiments are presented. First, the robustness and efficiency of walking motions, resulting from enforcing the virtual constraints of Section 4.1 through a feedforward plus PD controller developed in Section 4.6.1, is evaluated. Then, to achieve fast walking motions, the full compliant HZD controller developed in Section 4.6.2 is implemented.

5.1 Experiments

For each experiment, the controller was first coded in C++ and evaluated on a detailed simulation model of the robot that included encoder quantization and numerical estimation of velocity variables from encoder measurements. As discussed in Section 4.6.3, the controller was tested under various model perturbations, such as errors in the torso mass, spring stiffness, torso center of mass position, and deviations in initial conditions. The simulation model was then replaced with the physical robot. The experimental protocol is identical to the one used in [132, Sect. 4]. The experiments varied in duration from 78 steps to 265 steps, and were ended in each case by the experimenter stopping the robot and killing the power.

The results of the experiments are presented in Figures 5.2-5.10. In order to facilitate comparisons, Figures 5.2-5.4 assemble results from Exp. 1, 4 and 5; the remaining figures

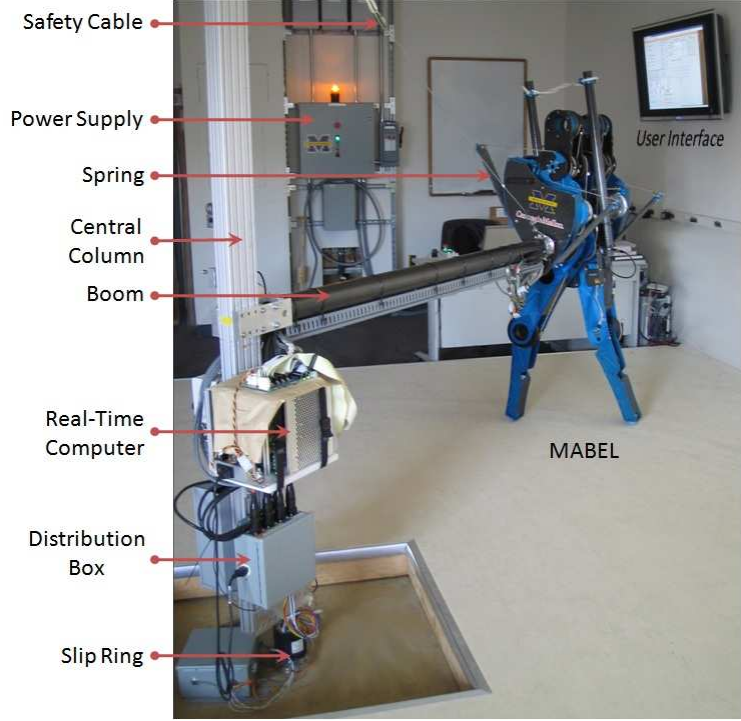


Figure 5.1: Experimental setup of the bipedal testbed MABEL.

pertain to individual experiments. In the experiments, the left leg refers to the inner leg, which is closer to the center boom, and the right leg refers to the outer leg, which is farther from the center boom. All walking speeds are measured with respect to the center point of the hip between the two legs. Videos of the experiments are available on YouTube [45].

5.1.1 Exp. 1: Nominal Walking at a Fixed Speed

This experiment approximately implements the virtual constraints depicted in Figure 4.4 with the feedforward plus PD controller given in (4.22). It was noticed in early experiments that the transition from the sc phase to the si phase given by (4.16) did not always occur. The spring was not decompressing to the 5° trigger point, and was probably due to the initial few steps being far away from the nominal orbit, and also because of inability of the controller to accurately track the stance motor leg shape virtual constraint. To ensure that the transition from the sc phase to the si phase always occurred in the experiments, the switching surface for this transition is modified to have a guard around the nominal value of θ_{sc}^- , such that the transition is guaranteed to occur for $\theta \in (\theta_{sc}^- - \gamma, \theta_{sc}^- + \gamma]$, with γ being

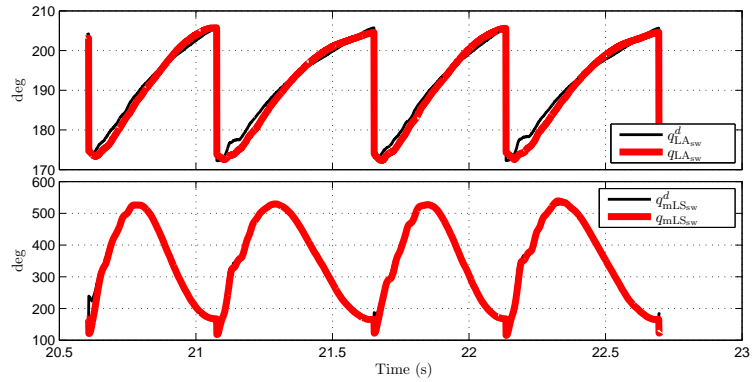
a small positive quantity. The modified switching surface is

$$\begin{aligned} \mathcal{S}_{sc \rightarrow si}^{exp} := & (\mathcal{S}_{sc \rightarrow si} \cap \{x_s \in TQ_s \mid \theta \in \\ & (\theta^{sc^-} - \gamma, \theta^{sc^-} + \gamma)\}) \cup \\ & \{x_s \in TQ_s \mid \theta = \theta^{sc^-} + \gamma\} \end{aligned} \quad (5.1)$$

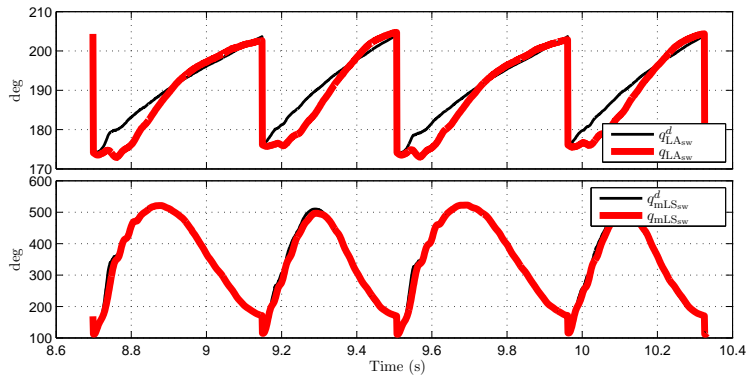
This controller led to successful walking, as illustrated in Figures 5.2 through 5.6. The evolution of the desired and achieved virtual constraints is depicted for the swing leg variables in Figure 5.2(a) and for the torso and stance leg motor positions in Figure 5.3(a). The nominal tracking in the swing leg is very good, whereas considerable errors occur in the torso and stance leg motor position. This is consistent with the fact that the swing leg is unloaded and lightweight, and hence much easier to control. The torques are given in Figure 5.4(a).

Figure 5.5 depicts the evolution of the torso angle and the evolution of B_{spring} for the left and right legs in stance, respectively, over 52 of the 78 steps in the experiment. Each of these is compared to the nominal fixed point. There is a pronounced asymmetry in the robot, as was noted in [49]. This asymmetry is due to the boom radius not being large enough and is currently not included as part of the model. It is also evident that the experimental gait is faster than the designed value: the nominal fixed point is 0.8 m/s, whereas the average experimental speed is approximately 1.0 m/s. One possible reason for the speed discrepancy is the impact model; see [132, p. 569]. This is discussed in Section 5.2. Another possible reason is the large errors in tracking the virtual constraints. As will be seen, controllers in subsequent experiments will reduce these errors and the walking speed will be closer to that of the fixed point.

Finally, Figure 5.6 illustrates the evolution of the B_{spring} -pulley for the stance and swing legs. Notice that the $sc \rightarrow si$ transition does not occur at five degrees as on the nominal orbit, and that in the swing phase, B_{spring} is not fixed at zero as assumed in the model.

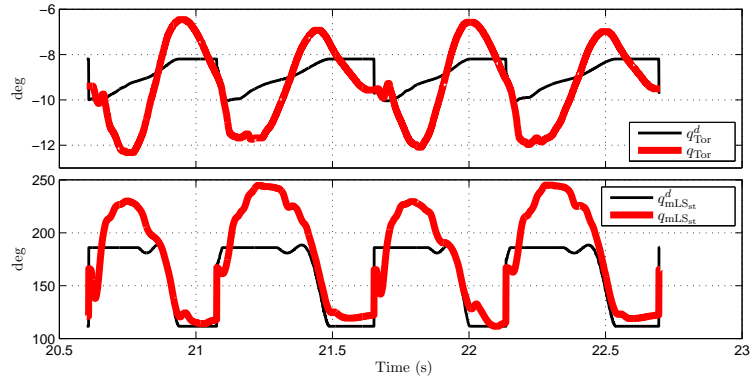


(a)

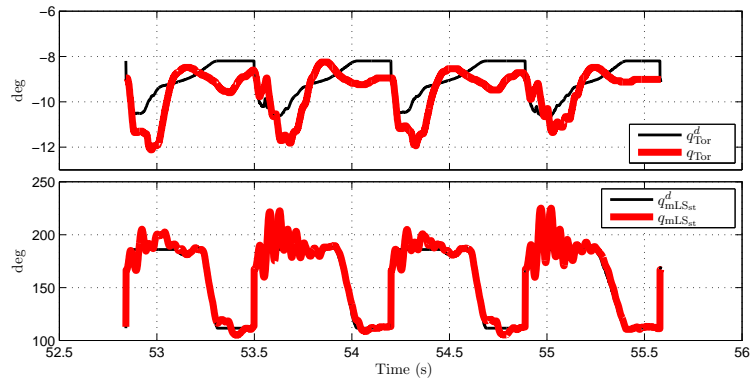


(b)

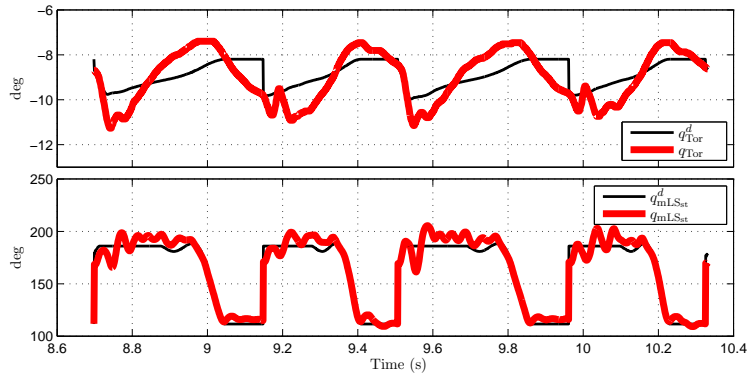
Figure 5.2: Tracking for the swing-leg virtual constraints for (a) PD + Feedforward controller in Exp. 1, and (b) Decentralized zero dynamics controller in Exp. 5. The tracking for PD + Feedforward controller is quite good, whereas the tracking for decentralized zero dynamics controller is worse right after impact and recovers quite well near impact.



(a)

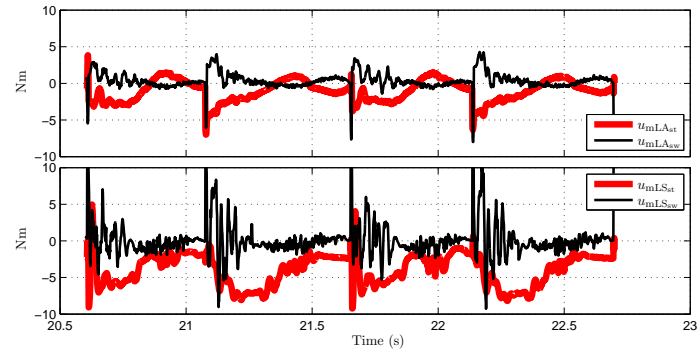


(b)

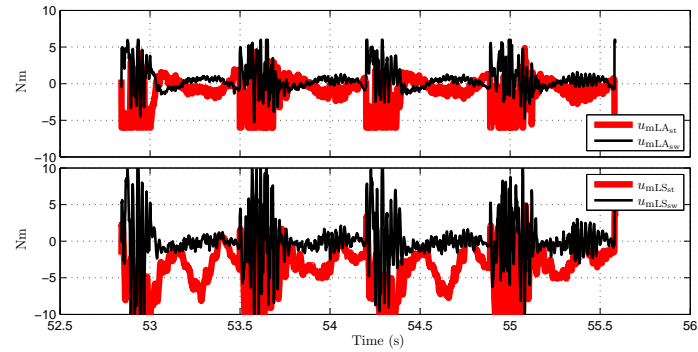


(c)

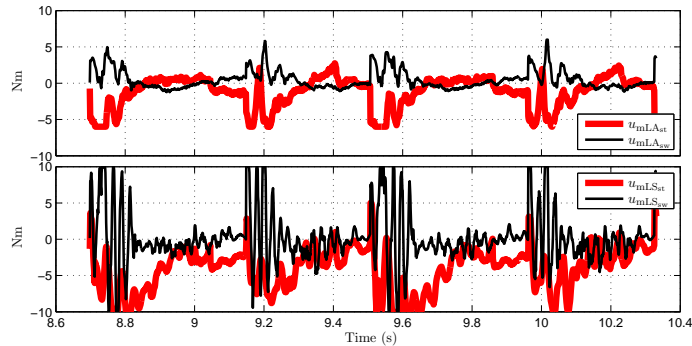
Figure 5.3: Tracking for the stance-leg virtual constraints for (a) PD + Feedforward controller in Exp. 1, (b) Compliant zero dynamics controller in Exp. 4, and (c) Decentralized zero dynamics controller in Exp. 5. The PD + Feedforward controller produces significant tracking errors. The tracking fidelity is much improved using the compliant zero dynamics controller. However, there are significant oscillations in tracking the motor leg shape, corresponding to a peak variation of approximately 1.3° in q_{LS} . The decentralized zero dynamics controller improves the tracking even further, with the oscillations nearly eliminated.



(a)



(b)



(c)

Figure 5.4: Motor torques for (a) PD + Feedforward controller in Exp. 1, (b) Compliant zero dynamics controller in Exp. 4, and (c) Decentralized zero dynamics controller in Exp. 5. The torques for the PD + Feedforward controller are noisy, but are comparable in magnitude to the nominal predicted values presented in Figure 4.6. The torques for the compliant zero dynamics controller appear more ‘noisy’ and experience significant saturation (saturation limits were set to 6 Nm on leg angle motors and 10 Nm on leg-shape motors). For the decentralized zero dynamics controller, the leg angle motor torques are far less ‘noisy’ than those for the compliant zero dynamics controller, while the leg shape motor torques are still a little ‘noisy’, but still much reduced when compared to the compliant zero dynamics controller.

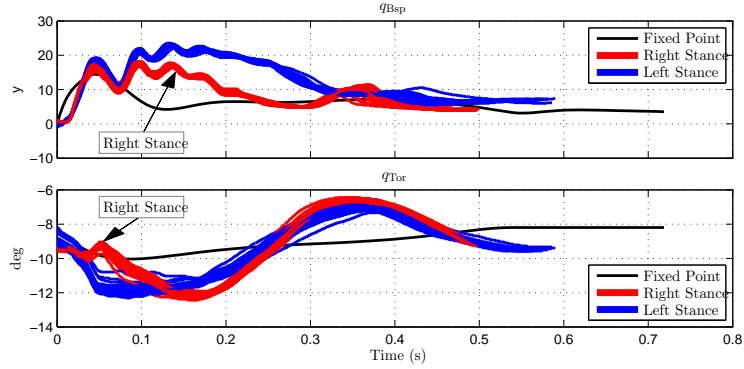


Figure 5.5: This figure compares the evolution of $q_{B_{sp}}$ for the left and right legs in stance for Exp. 1 and the evolution of the torso, over 52 steps, and compares them with the corresponding values for the nominal fixed point. The step times for the left and right legs vary, with the robot walking faster when the right (i.e., inside) leg is stance, and both step times are shorter than the fixed point. This is evident when we compare the average walking speed in the experiment, 1.0 m/s, with the designed fixed point walking speed of 0.8 m/s. Possible reasons for this discrepancy are discussed in the text. Next notice in the $q_{B_{sp}}$ plot, right after impact, in the experiments the spring compresses more slowly than in the fixed point. This is possibly due to a non-instantaneous double support phase in the experiments. (The duration of the double support in the walking experiments is around 20 ms.)

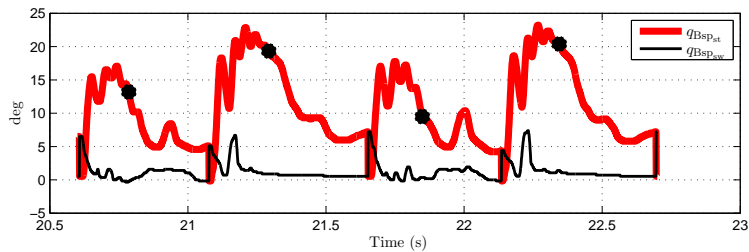


Figure 5.6: Stance and swing B_{spring} evolution for nominal walking experiment. The asterisks indicate the locations of the transitions from subphase sc to subphase si . This transition occurs before the $q_{B_{sp}} = 5^\circ$ event due to the presence of the additional guards in the experiments, (5.1). It is also seen that the swing B_{spring} angle is not at 0° as assumed in the model. This deviation is significant right after impact, but appears to quickly die out to a small value for the remaining part of the gait.

5.1.2 Exp. 2: Demonstration of Robustness to Perturbations

To test the robustness of the controller used in Exp. 1, external, short duration forces are applied at the hip at various instants of time. The results are depicted in Figure 5.7. Initially, the robot is pushed forward by the experimenter, causing the robot to speed up by roughly 45%. During the subsequent ten steps, the speed slowly converges back to the nominal. Next, a retarding force is applied, causing the robot to slow down by roughly 45%. Over the next three steps, the robot's speed has essentially returned to the unperturbed value. This experiment demonstrates the robustness of the robot in closed loop with the feedback controller given by (4.22) to external disturbances and illustrates an asymmetry in the rejection of the speed perturbation. A simulation of the model was carried out to estimate the force applied. A constant force over the second half of the gait was assumed to be applied at the end of the boom. In order to achieve similar speed gains and speed drops, the required force from simulation is around 78 N in the forward direction, and around 71 N in the reverse direction, respectively.

When the robot is pushed forward (external energy is injected into the system), the speed of the robot increases, and the robot takes a large number of steps to recover. One would expect that, due to the increased speed, larger amounts of energy would be dissipated at impacts after the forward push perturbation. This would cause the robot to converge to its nominal motion faster. However, this is not the case in the experiment, indicating that the energy loss at impact is fairly small. Thus a large number of steps are needed to dissipate the extra energy injected into the system by the push. This is also what we see in simulation in Section 5.2.2. This indicates that the compliance plays an important role in the impacts. For pushes in the opposite direction, additional energy is quickly injected into the system by the actuators.

5.1.3 Exp. 3: Efficient Walking

The fixed point obtained by optimizing for the specific energetic cost of mechanical transport (4.19) is implemented experimentally using the feedforward plus PD controller (4.22). The designed fixed point has a cost of $c_{mt} = 0.0385$ and nominal walking speed of

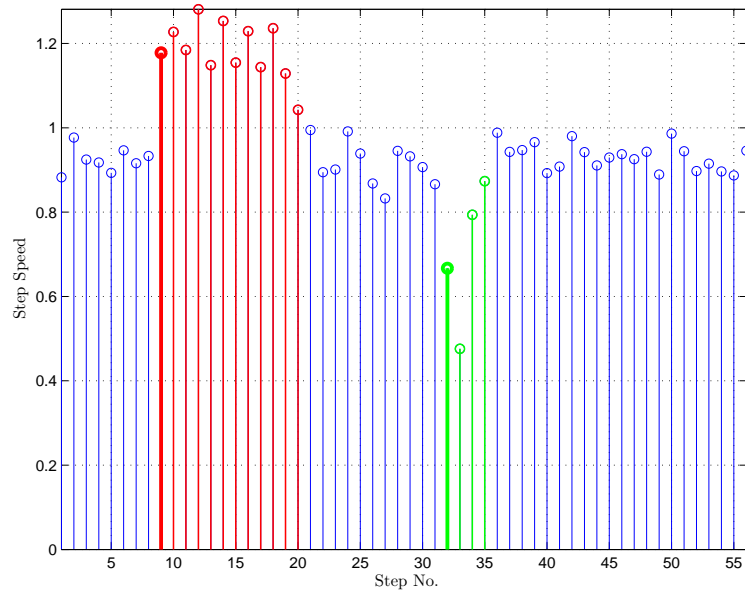


Figure 5.7: Speed at each step for Exp. 2. An external disturbance in the form of a forward push is applied on step 9 (thick line), and the speed of the subsequent ten steps is faster than the nominal. An external disturbance in the form of a backward push is applied on step 32 (thick line), and the speed of the subsequent three steps is slower than the nominal. In both cases nominal speed is recovered. Interestingly, it takes more steps for the robot to lose energy and slow down rather than for the robot to gain energy and speed up. This indicates that very little energy is lost at impact.

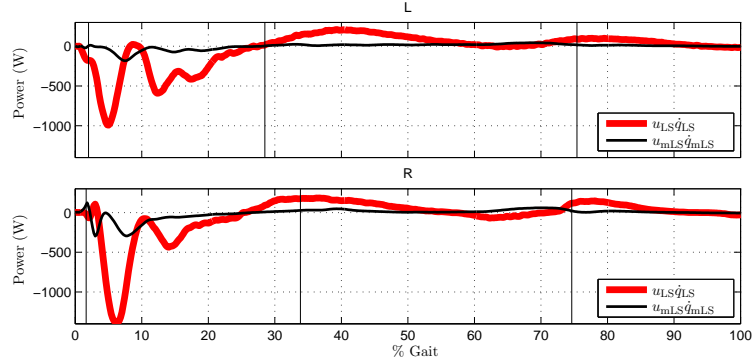


Figure 5.8: Power plot of Exp. 3 implementing a fixed point obtained by optimizing for c_{mt} (4.19). The plots are obtained by averaging the power over 77 steps (39 steps with the left leg as stance and 38 steps with the right as stance). The vertical lines indicate mean phase transition instants. Most of the negative work is done by the compliance instead of the actuator.

0.8 m/s.

Figure 5.8 depicts the power plots for the inside and outside legs obtained by averaging the experimental data over 77 steps. The realized energetic cost of mechanical transport is $c_{mt} = 0.14$ ¹. For comparison purposes, Figure 5.9 shows the power plot for the hand-tuned virtual constraints reported in [49]. It is clear that the new control design makes more efficient use of the compliance available in the open-loop plant than the controllers previously designed.

The experimentally realized c_{mt} is approximately three times the designed value. This is discussed in Section 5.2.3.

5.1.4 Exp. 4: Compliant Zero Dynamics Controller

This section focuses on achieving greater fidelity in the tracking of the virtual constraints.

Background: An important goal of MABEL is running. As an intermediate goal, we have started looking at fast walking. We consequently designed new fixed points using the methods of Sections 4.1 and 4.4, for walking at 1.0 and 1.2 m/s, and implemented them using the controller structure of (4.22). Experiments were unsuccessful, even when the transition controller of [130, Chap. 7] was added. The experimental data showed poor

¹This is around 10% lower than the realized energetic cost of mechanical transport for Exp. 1, which was $c_{mt} = 0.15$. The designed values of the energetic cost of mechanical transport for the corresponding fixed points also differ by the same amount.

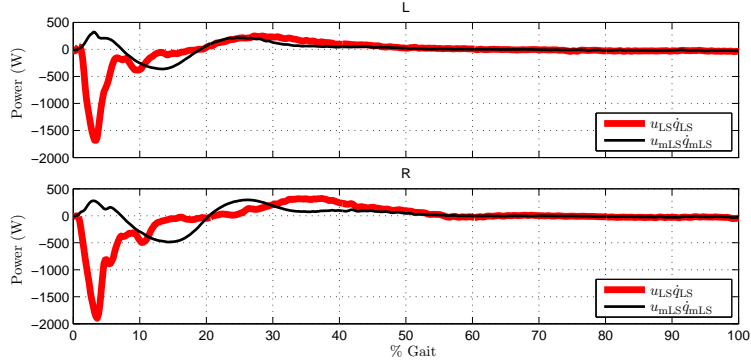


Figure 5.9: Power plot for the hand-tuned virtual constraints experiment reported in [49]. The plots are obtained by averaging the power over 32 steps. Immediately after impact and during the first 10% of the gait, we can see the motor actually fights the spring. In the rest of the gait, the motors do almost all the work.

tracking of the torso and stance motor leg shape virtual constraints and led us to conclude that more precise control was needed in order to achieve these higher speeds.

We thus stepped away momentarily from the problem of fast walking and concentrated on achieving a higher fidelity implementation of the virtual constraints. We decided to use the full I/O linearizing controller (4.21), with correction polynomials as in (4.24). Although zero dynamics based controllers are great in theory, all experimental implementations to date had been with simple PD controllers [132, 85]. Zero dynamics controllers are often criticized for being overly dependent on the model being accurate, and for being too complex to implement in real time.

Results: We report, for the first time, an experimental implementation² of the full compliant hybrid zero dynamics controller to successfully achieve walking on MABEL. The virtual constraints of Exp. 1 are used here. The tracking accuracy obtained is far better than the feedforward plus PD controller used previously. The compliant zero dynamics controller, (4.11), (4.21), with the correction terms, (4.24), and with the modified event transition surface, (5.1), is deployed. The output coordinates are normalized to approximately the same magnitude for better conditioning of the decoupling matrix. Unlike in Section 5.1.1, u^* is computed from the dynamics directly and is not approximated. (Recall that in Section 4.6.1, u^* was regressed against θ to obtain a Bézier polynomial $u^*(\theta, \beta)$). This approximated

²To enable computing all terms of the zero dynamics based controller within a 1ms sample time, an extremely efficient matrix library based on C++ expression templates [123, 5] was used.

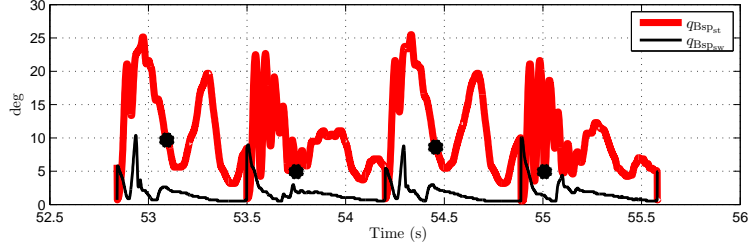


Figure 5.10: B_{spring} evolution for the compliant zero dynamics controller in Exp. 4. The asterisks indicate location of transition from the stance-compression (sc) to the stance-decompression (sd) phase. The transitions appear to be closer to the nominal value of $q_{\text{BSP}} = 5^\circ$ when compared to the PD controller, Figure 5.6.

u^* along the periodic orbit as a function of θ but provides no velocity correction.)

The tracking of the swing virtual constraints is at least as good as that obtained in Figure 5.2(a) with the PD controller and is not shown. Figures 5.3(a) and 5.3(b) compare the tracking of q_{Tor} and $q_{\text{mLS}_{\text{st}}}$ under the effect of the PD and the compliant zero dynamics controllers, respectively. The tracking is greatly improved, where the error on q_{Tor} reduced from a peak of 3.1° to 2.4° , with a reduction in root mean square error (RMSE) from 1.44° to 0.89° . Similarly, the error on $q_{\text{mLS}_{\text{st}}}$ reduced from a peak of 59.1° to 39.5° , with a reduction in RMSE from 29.82° to 0.28° (when scaled by a gear ratio of 31.42, this translates to a reduction in error for $q_{\text{LS}_{\text{st}}}$ from a peak of 1.9° to 1.3° , with a reduction in RMSE from 0.95° to 0.009° .)

Figure 5.10 illustrates the stance and swing B_{spring} values over a few steps. Notice that the $sc \rightarrow si$ transition occurs more closely to the designed value. Figure 5.4(b) depicts the control torques at the actuators. These are noisier than with the PD control torques. This issue will be resolved in the next section.

The average walking speed for this experiment is 0.9 m/s. This is closer to the designed walking speed than in Exp. 1. The reduction in errors in the virtual constraint improves the correlation between the desired and realized walking speeds.

5.1.5 Exp. 5: Fast Walking

We return to the problem of achieving fast walking. A *decentralized* zero dynamics controller is implemented using the virtual constraints of Exp. 1. This is simply the zero

dynamics controller as implemented in Exp. 4, with the off-diagonal elements of the decoupling matrix, $L_{g_s}L_{f_s}h$, set to zero. This was observed to reduce the ‘cross-talk’ in the control signal due to errors in one output being transferred to another. Figures 5.2(b) and 5.3(c) illustrate the virtual constraint tracking achieved with the decentralized zero dynamics controller. Figure 5.4 compares the torques obtained under the effect of the presented controllers. The torques for the decentralized zero dynamics controller are less noisy when compared to the torques for the compliant zero dynamics controller.

With this controller, MABEL started walking at around 1.15 m/s. The torso was gradually leaned forward to increase the speed. A top walking speed of 1.52 m/s was achieved with a sustained walking speed of 1.5 m/s (3.4 mph.) This made MABEL “the fastest walking biped of any size” in the world on October 31, 2009, and the record was held until April 22, 2010, when PETMAN [94] reclaimed the speed record with a 1.97 m/s walking gait; the video was posted to YouTube [45].

Section 5.2.4 compares the walking speed of several bipedal robots.

5.2 Discussion of the Experiments

This section discusses various aspects of the robot and the feedback controllers revealed by the experiments.

5.2.1 Asymmetry

The model used in the feedback designs has assumed a planar robot; in particular, this is predicated on the robot’s dynamics being identical when the left leg is in stance or the right leg is in stance. The data shows clearly that this is false. The robot itself is nearly symmetric. The asymmetry arises from the boom used to constrain the robot to the sagittal plane. The facility housing the robot only permits a boom of length 2.25 m from the center of the floor to the center of the robot. The width of MABEL’s hips is 0.24 m, which is approximately 10% of the boom. For comparison, RABBIT has a boom of length 1.7 m, with a hip width of 0.074 m, which is approximately 4% of the boom. The robot plus boom may need to be modeled as a 3D system.

It was noted that the experimental walking speed for Exp. 1 in Section 5.1.1 was 1.0 m/s. The average speed with the left (inner) leg as stance is 0.98 m/s while the average speed with the right (outer) leg as stance is 1.02 m/s. The ratio of left and right stance speeds is nearly equal to the ratio of distance of left and right legs to the center column.

5.2.2 Impact Model

For legged robots, the accuracy of the model of the leg end (foot) impact with the walking surface is difficult to ascertain and to improve. The vast majority of researchers adopt an instantaneous double support model, and use [57] to build the corresponding mathematical model. Several researchers have used or proposed compliant ground models [130, p. 278], [89, 13, 33, 12, 11, 88, 127, 128], [100, Chap. 5] yielding a double support phase of nonzero duration. Results exist in the literature in which rigid impact models have also been used to obtain non-trivial double support phases [80, 99]. While the compliant models seem more physically realistic, the uncertainty present in the parameters of such models does not necessarily yield a more accurate result, and it certainly does add considerable complexity to the model via numerical stiffness and / or non-Lipschitz continuous dynamics.

The bottom line is that when comparing theoretical predictions to experiments, the impact model should be considered as one possible source of error. In all of the experiments reported here, the robot walked faster than predicted by the dynamic model. The reference [132, Fig. 9] suggested that this could be accounted for by scaling the post-impact velocity predicted by (3.19). The scaling is performed here in such a way that the post-impact velocity still respects the constraints (3.22), (3.23), (3.24). The impact scaling is achieved by replacing \dot{q}_e^+ with $\eta\dot{q}_e^+$ in these constraints, where η is an impact scaling factor. Table 5.1 shows various steady-state walking speeds for different values of the impact scaling factor. It is notable that for the simulation to match the experimental walking speed, we require $\eta = 0.966$, a change of less than 4% to the impact map. Figure 5.11 compares the nominal walking experiment with a simulation with this impact scaling factor.

Table 5.1: Effect of Impact map scaling on walking speed.

| Impact map scaling factor η | Steady-state walking speed |
|----------------------------------|----------------------------|
| 1.0 | 0.80 m/s |
| 0.99 | 0.86 m/s |
| 0.98 | 0.92 m/s |
| 0.97 | 0.98 m/s |
| 0.966 | 1.00 m/s |
| 0.96 | 1.04 m/s |

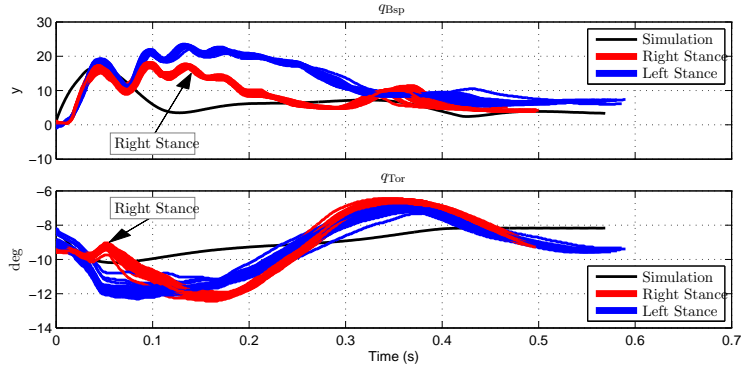


Figure 5.11: This figure shows the plots of the B_{spring} and the Torso separately for the left and right legs as stance for Exp. 1 and compares with a simulation of the nominal fixed point with an impact scaling factor, $\eta = 0.966$.

5.2.3 Exp. 3: Efficient Walking

As mentioned in Section 5.1.3, the experimentally realized c_{mt} is approximately three times the designed value.

One may suppose that this discrepancy is due to the existence of non-negligible motion of the motor shaft during the sc and sd phases, a consequence of the inability of the actuator to function as an ideal brake. However, this is not the case. Detailed examination of the mechanical work performed in the sc and sd phases reveals that the stance motor leg-shape already does little work in the experiment. Introducing a brake in the experimental setup would then have little effect in improving the c_{mt} value. After several simulations, perturbing different parts of the model, such as increasing friction and introducing cable stretch, we have observed that both friction and cable stretch contribute significantly to increase the specific cost of mechanical transport.

Table 5.2 illustrates the (mechanical) energy efficiency of several bipedal robots, and is sorted by c_{mt} . Although the experimentally obtained c_{mt} value for MABEL is not as

Table 5.2: Efficiency numbers for various bipedal robots and various controllers on MABEL.

| Robot | c_{mt} |
|---------------------------------------|----------------------------|
| Honda's ASIMO* | 1.60 |
| RABBIT [129, Sec. 6.5.1] | 0.38 |
| MABEL - Hand designed VC [49] | 0.29 |
| MABEL - 0.8 m/s FP, HZD Ctrl (Exp. 4) | 0.18 |
| MABEL - 0.8 m/s FP (Exp. 1) | 0.15 |
| MABEL - $c_{mt} = 0.0385$ FP (Exp. 3) | 0.14 |
| T.U. Delft's Denise* [27] | 0.08 |
| MIT's Spring Flamingo [27] | 0.07 |
| Cornell Biped* [26] | 0.055 |
| McGeer's Dynamite [27] | 0.04 |

*3D, autonomous and untethered.

good as the designed value, it is 12 times better than that of Honda's ASIMO, over 2.75 times better than RABBIT and twice better than the hand-tuned virtual constraints based controller on MABEL. This puts MABEL's energy efficiency within a factor of two of T.U. Delft's Denise and a factor of three of the Cornell Biped, none of which can step over obstacles or run; it is also within a factor of two of the MIT Spring Flamingo which can easily step over obstacles but cannot run, and within a factor of three of humans, who can do all of the above.







5.2.4 Exp. 5: Fast Walking

In order to compare MABEL's walking performance with other bipedal robot designs and control methods, Table 5.3 lists robot parameters, peak walking speed, and the dimensionless velocity³. The table is sorted by peak speed. Of note is the bipedal robot RunBot [76, 36], which is the fastest walker measured by dimensionless velocity and leg lengths per second. MABEL was the fastest walker in terms of absolute speed from October 31, 2009 until April 22, 2010, when PETMAN took the record.

Notice that MABEL and its predecessor, RABBIT, are the only ones in this list without ankles and feet. It has been suggested in [74] that ankles and even toes, in humans, are very useful to provide a push-off to increase speed. The effect of push-off in bipedal robots

³The dimensionless velocity serves as a speed metric and is defined as the square root of the Froude number, with the Froude number being the ratio of the centrifugal force due to motion about the foot and the weight of the robot [122].

Table 5.3: Top walking speeds of bipedal robots; this table is motivated by [76, Figure 1].

| |  |  |  |  |  |  |
|------------------------|---|---|---|--|---|---|
| | RunBot | RABBIT | Spring Flamingo | MABEL | PETMAN | Olympic Record |
| Mass (Kg) | 0.53 | 32.00 | 14.20 | 65.00 | - | ≈ 70.00 |
| Leg Length (m) | 0.23 | 0.80 | 0.90 | 1.00 | - | 0.9 ≈ 1.15 |
| Peak Speed (m/s) | 0.80 | 1.20 | 1.25 | 1.50 | 1.97 | 4.60 |
| Dimensionless Velocity | 0.53 | 0.43 | 0.42 | 0.48 | - | 1.4 ≈ 1.5 |

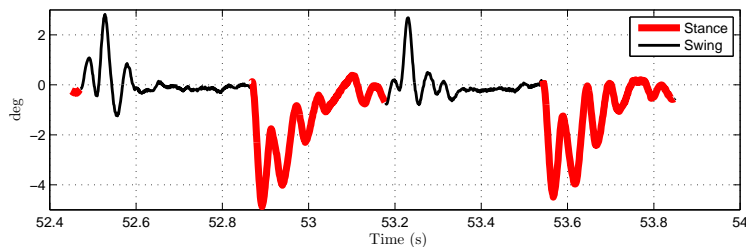


Figure 5.12: Cable stretch as measured by $q_{LA} - q_{mLA}/23.53$ for the right (outer) leg over two consecutive swing and stance phases for the fast walking experiment, Exp. 5. Negative cable stretch values are not to be interpreted as cable compression. Cable differentials require two sets of cables to ‘pull’ in either direction. The negative cable stretch values in the figure refer to cable stretch in the second cable.

is studied in [71] and is established as an energy efficient way to increase speed.

5.2.5 Cable Stretch

The differentials in MABEL’s drivetrain, c.f. Figure 3.1, are realized by a series of cables and pulleys. The reader is referred to [58, 50] for details. The robot was designed under the assumption that the cables undergo zero deformation, and this assumption has been used in developing the dynamic model of the robot that we used for control design. In the experiments, it has been observed that there is significant cable stretch. A representative plot of the cable stretch in the leg angle is shown in Figure 5.12, where the variable $q_{LA} + q_{mLA}/23.53$ is plotted in degrees. If the cables were rigid, this variable would be identically zero⁴.

⁴Encoders are present to directly measure both q_{LA} and q_{mLA} .

5.2.6 Zeroing the Virtual Constraints

The theorems supporting the method of virtual constraints are easier to prove when an input-output linearizing controller such as (4.21) is used [84, 130, 92]. In practice, the benefits are achieved by any controller that realizes the constraints $h(q_s) = H_0 q_s - h_d(\theta)$ with “sufficient accuracy”. This can be formalized using high-gain feedback via singular perturbations [124]. On RABBIT the actuators were connected to the robot’s linkage through 50 : 1 gear ratios, and it turned out that a high-gain PD implementation resulted in sufficiently accurate realization of the virtual constraints. However, on MABEL, this simple controller was inadequate for fast gaits, motivating the implementation of an input-output linearizing controller. We believe that three things limited our ability to increase the PD gains in (4.22) sufficiently high to realize the constraints: lower gear ratios in the drivetrain⁵; encoder ‘noise’ when estimating derivatives; and the additional compliance arising from the cable stretch discussed above.

⁵MABEL has a gear ratio of 23.53 : 1 and 31.42 : 1 for leg angle and leg-shape coordinates respectively. For comparison, RABBIT has a gear ratio of 50 : 1 at both the knees and hips.

CHAPTER VI

Control Design for Running: Embedding Active Force Control within the Compliant Hybrid Zero Dynamics

This chapter presents a controller for inducing stable running motions on MABEL. The controller will create an actuated compliant HZD, and enable active force control within the HZD.

Similar to walking, a set of virtual constraints is chosen so that the open-loop compliance of the system is preserved as a dominant characteristic of the closed-loop system. In addition, active force control will be introduced as a means of varying the effective compliance of the system. The motivation for this control approach is elaborated in Section 6.1.

This chapter is organized as follows. Section 6.1 motivates the control design for embedding active force control within the compliant HZD framework. Section 6.2 will present a high-level overview of the control design. Section 6.3 presents the virtual constraints for the stance phase of running that result in a restricted dynamics that is compliant and actuated; the associated zero dynamics are given in Section 6.4. Section 6.5 specifies the controller used for the active force control which provides a means of varying the effective compliance of the system. Section 6.6 presents the virtual constraints for the flight phase of running and Section 6.7 presents the associated zero dynamics. Section 6.8 presents the hybrid restricted dynamics model. Section 6.9 presents the optimization process for gait design. Section 6.10 presents one fixed point representing a periodic running motion. Section 6.11 presents the closed-loop control design. Section 6.12 presents additional controller modifications that

need to be done prior to experimental deployment.

6.1 Motivation for Control Design

In the walking experiments reported in Chapter V, the spring on the stance leg compresses on impact and decompresses nearly to its rest position within 100 ms. This property is dependent on the mass of the robot and spring stiffness. Using the same set of springs for running, and with the control strategy of holding the motor position constant at impact, would yield stance times of around 100 ms. Since there is no control authority on the torso in the flight due to the conservation of angular momentum, any errors on the torso positions have to be corrected during the stance time. Feedback to correct the potentially large errors for the torso within 100 ms would place large torque requirements on the actuators and would potentially be infeasible.

Hence, longer stance times are necessary. One solution to obtain longer stance times would be to reduce the spring stiffness by physically replacing the springs present in MABEL with softer springs. However, as investigated in Rummel and Seyfarth [101], having compliance in the joint level with segmented legs results in a nonlinear relationship between leg compression and leg force. Thus, reducing the spring stiffness on MABEL would have the effect of the robot collapsing at moderate leg compressions owing to the fact that the less stiff spring is not able to provide sufficient leg force to hold up the robot. This would significantly reduce the range of impact angles for the knee for which the springs could hold up the weight of the robot. Thus, there is a need to vary the effective compliance of the leg in different parts of the stance phase without resorting to softer springs.

We look now at inspiration from biomechanical studies. Ferris et al., [31, 32] carried out experiments on human runners and found that runners adjust their leg stiffness to accommodate for variations in surface stiffness, allowing them to maintain similar running mechanics (e.g., peak ground reaction force and ground contact time) on different surfaces. Moreover, they suggest that incorporating an adjustable leg stiffness in the design of running robots is important if they are to match the agility and speed of animals on varied terrain. Further, in a set of impressive experiments carried out by Daley et al., [29, 28], where guinea

fowl are subjected to large unexpected variations in ground terrain, it is suggested that the animals can accommodate this variation in ground height by varying their leg stiffness.

In summary, there is a need for a control strategy which can dynamically vary the effective compliance of the leg. In addition, active force control has been suggested as a way to increase robustness to perturbations in ground height and ground stiffness in [70]. In the following sections, we develop a controller based on virtual constraints and the framework of hybrid zero dynamics to have the capability of dynamically varying the effective leg stiffness.

6.2 Overview of the Control Method

The control objective is to design a periodic running gait that is exponentially stable and has a sufficiently large domain of attraction so as to accommodate inevitable differences between the model and the robot. Virtual constraints are used to impose constraints on the robot's dynamics in the stance and flight phases. By a judicious choice of variables on which the constraints are to be imposed, the resulting restricted stance dynamics is made compliant and actuated. The input in the zero dynamics for the stance phase is used to change the effective compliance of the robot. Discrete-event-based control is then employed to (a) create hybrid invariance, (b) exponentially stabilize the periodic gait, and (c) increase the domain of attraction of the periodic gait.

To achieve the control objectives, the feedback controller introduces control on three levels. Figure 6.1 depicts the overall structure of the running controller. On the first level, continuous-time feedback controllers Γ_p^α with $p \in \mathcal{P} := \{\text{s}, \text{f}\}$ are employed in the stance and flight phases to create invariant and attractive surfaces embedded in the state space for each of the respective phases. The discrete-time feedback controllers $\Gamma_p^{\alpha_c}$ are employed in the transitions between the phases in order to render these surfaces hybrid invariant. For later reference, these surfaces are $\mathcal{Z}_{(\alpha_p, \alpha_c^p, \beta, \gamma)} \subset TQ_p$.

On the second level, an event-based controller Γ^β performs step-to-step parameter updates to render the periodic orbit, representing running and embedded in these surfaces, exponentially stable. Finally, on the third level, another event-based controller Γ^γ performs

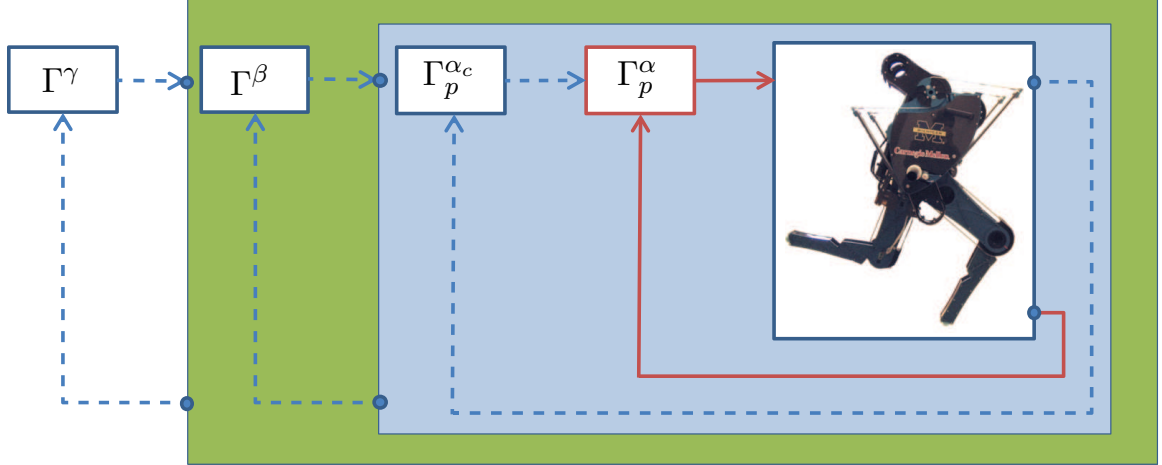


Figure 6.1: Feedback diagram illustrating the running controller structure. Continuous lines represent signals in continuous time; dashed lines represent signals in discrete time. The controllers Γ_p^α and $\Gamma_p^{\alpha_c}$ create a compliant actuated hybrid zero dynamics. The controller Γ^β ensures that the periodic orbit on the resulting zero dynamics manifold is locally exponentially stable. The controller Γ^γ improves the domain of attraction of the periodic orbit.

step-to-step parameter updates to increase the domain of attraction of the periodic orbit.

The remaining sections of this chapter will develop the procedure described above in greater detail and make it mathematically precise.

6.3 Virtual Constraint Design for Stance

Recall that virtual constraints are holonomic constraints on the robot’s configuration variables that are asymptotically imposed through feedback control. They were used in Section 4.1 to synchronize the evolution of the robot’s links for synthesizing walking gaits on MABEL. For walking, one virtual constraint was designed per independent actuator.

For the stance phase of running, the virtual constraints are parametrized by θ_s , a strictly monotonic function of the joint configuration variables. As in walking, we choose θ_s to be the absolute angle formed by the virtual compliant leg relative to the ground (see Figure 4.1), i.e.,

$$\theta_s(q_s) = \pi - q_{LA_{st}} - q_{Tor}. \quad (6.1)$$

The virtual constraints for the stance phase can then be expressed in the form

$$y_s = h_s(q_s) = H_0^s q_s - h_d^s(\theta_s), \quad (6.2)$$

We have two design choices to make: (a) The controlled variables $H_0^s q_s$, and (b) the constraints $h_d^s(\theta_s)$.

6.3.1 Deciding What to Control

As motivated in the previous section, we need some means of varying the effective compliance of the system for designing running gaits that are robust and have stance phases that are sufficiently long that torso correction is feasible. Since the transmission of MABEL places a spring in series with the leg shape actuator for the stance leg, $u_{\text{mLS}_{\text{st}}}$, force control on this actuator can be employed to vary the effective compliance of the system. To achieve this, we choose to impose virtual constraints on three controlled variables using three of the actuators and leave the stance motor leg shape actuator for active force control instead of using it for imposing an additional virtual constraint. This increases the dimension of the zero dynamics, which may seem counter-intuitive. Such a strategy was employed in the past by Choi and Grizzle [24] for the control of fully actuated planar bipeds with feet by choosing not to impose a virtual constraint using the ankle actuator. Another instance when one of the actuators was not employed in imposing a virtual constraint is in the work by Poulakakis and Grizzle [92] on ASLIP, where the leg force actuator is not used to enforce a virtual constraint, but rather used to achieve a target zero dynamics that is diffeomorphic to the dynamics of a SLIP.

By choosing to impose three virtual constraints, we have three control variables to specify. The torso is selected as one of the controlled variables (as was done for walking). Since the torso represents over 65% of the mass of the robot, the entry conditions for the torso into the flight phase are imperative for running. Due to the conservation of angular momentum, there is minimal control authority on the torso position in the flight phase. The initial conditions at the entry into the flight phase essentially determine the evolution of the torso in the flight phase. Next, on the swing leg, the controlled variables are as in

walking. In summary, the controlled variables are

$$H_0^s q_s = \begin{bmatrix} q_{LA_{sw}} \\ q_{mLS_{sw}} \\ q_{Tor} \end{bmatrix}. \quad (6.3)$$

6.3.2 Specification of the Constraints

The stance phase of running is broken into two subphases: stance-compression and stance-decompression. For the three virtual constraints, the two sub phases are combined into a single phase. Since all the virtual constraints are specified by combining both the subphases, the division of the stance phase is artificial with respect to the virtual constraints. The need for such a breakup is described more fully in Section 6.8. We use the framework of virtual constraints with subphases as developed for walking (see Appendix A.)

The virtual constraints are parametrized by 5th order Bézier polynomials. The desired trajectory of each of the controlled variables is denoted by $h_{LA_{sw}}^{d,s}$, $h_{mLS_{sw}}^{d,s}$, and $h_{Tor}^{d,s}$ respectively with corresponding Bézier coefficients $\alpha_{LA_{sw}}^s$, $\alpha_{mLS_{sw}}^s$, and α_{Tor}^s . The desired trajectories of the virtual constraints are assembled as

$$h_d^s(\theta_s, \alpha_s) = \begin{bmatrix} h_{LA_{sw}}^{d,s}(\theta_s, \alpha_s) \\ h_{mLS_{sw}}^{d,s}(\theta_s, \alpha_s) \\ h_{Tor}^{d,s}(\theta_s, \alpha_s) \end{bmatrix}, \quad (6.4)$$

where the Bézier coefficients are organized as

$$\alpha_s = \begin{bmatrix} \alpha_{LA_{sw}}^s \\ \alpha_{mLS_{sw}}^s \\ \alpha_{Tor}^s \end{bmatrix}. \quad (6.5)$$

Next we discuss the choice of the general shape of the virtual constraints. As mentioned earlier, the initial conditions of the heavy torso at the entry of the the flight phases is important for running. For running forward, we expect the torso to be leaning forward during most of the gait. If in addition, at the entry to the flight, the torso has a forward

velocity, then due to the conservation of angular momentum, the torso continues to move forward in the flight phase and on landing the torso may have excessively pitched forward. Correcting a large error in torso in the relatively small (compared to walking) stance phase is difficult. To prevent this, the virtual constraint for the torso is designed such that, at the end of the stance phase, the torso is leaning forward but has a backward velocity.

The swing leg virtual constraints are chosen such that the swing leg angle moves forward in the stance phase, and the swing motor leg shape lifts the leg higher to provide ground clearance of the swing leg. These constraints are similar to those designed for walking.

6.4 Stance Zero Dynamics

The open-loop stance-phase dynamics are given by (3.14). By a change of coordinates, the inputs to the system can be separated into two pairs - the stance motor leg shape input $u_{\text{mLS}_{\text{st}}}$ with input matrix $B_{\text{mLS}_{\text{st}}}$ and \tilde{u} representing the inputs excluding the stance motor leg shape input, with corresponding input matrix \tilde{B} . With this, the open-loop dynamics can be written in the standard form as,

$$D_s(q_s) \ddot{q}_s + H_s(q_s, \dot{q}_s) = B_{\text{mLS}_{\text{st}}} u_{\text{mLS}_{\text{st}}} + \tilde{B} \tilde{u}, \quad (6.6)$$

where H_s is as in (3.14). By the choice of the virtual constraints, specifically choosing only three control variables on which the constraints are imposed, the stance motor leg shape input remains free and is not used for imposing a virtual constraint. We intend to implement active force control on the stance motor leg shape input, in which case, this becomes a function of the state, and with an abuse of notation we can write $u_{\text{mLS}_{\text{st}}} = u_{\text{mLS}_{\text{st}}}(x_s)$. With this, the state space representation of the stance dynamics with active force control on the stance motor leg shape input is

$$\begin{aligned} \dot{x}_s &= \begin{bmatrix} \dot{q}_s \\ -D_s^{-1} H_s + D_s^{-1} B_{\text{mLS}_{\text{st}}} u_{\text{mLS}_{\text{st}}} \end{bmatrix} + \begin{bmatrix} 0 \\ D_s^{-1} \tilde{B} \end{bmatrix} \tilde{u} \\ &:= \tilde{f}_s(x_s) + \tilde{g}_s(x_s) \tilde{u}. \end{aligned} \quad (6.7)$$

As discussed in Section 6.3, an output function y_s has been associated with the continuous stance dynamics with active force control in (6.7). The zero dynamics is defined as the maximal internal dynamics of the system that is compatible with the output being identically zero [64]. Differentiating the output twice with respect to time results in

$$\frac{d^2 y_s}{dt^2} = L_{\tilde{f}_s}^2 h_s(x_s, \alpha_s) + L_{\tilde{g}_s} L_{\tilde{f}_s} h_s(q_s, \alpha_s) \tilde{u}, \quad (6.8)$$

where $L_{\tilde{g}_s} L_{\tilde{f}_s} h_s(q_s, \alpha_s)$, the decoupling matrix, has full rank. Under the conditions of [130, Lemma 5.1],

$$u_s^*(x_s, \alpha_s) := - \left(L_{\tilde{g}_s} L_{\tilde{f}_s} h_s(q_s, \alpha_s) \right)^{-1} L_{\tilde{f}_s}^2 h_s(x_s, \alpha_s), \quad (6.9)$$

is the unique control input that renders the smooth six-dimensional embedded submanifold

$$\mathcal{Z}_{\alpha_s} = \left\{ x_s \in TQ_s \mid h_s(q_s, \alpha_s) = 0, L_{\tilde{f}_s} h_s(x_s, \alpha_s) = 0 \right\} \quad (6.10)$$

invariant under the stance dynamics (6.7); that is, for every $z_s \in \mathcal{Z}_{\alpha_s}$,

$$f_s^*(z) := \tilde{f}_s(z) + \tilde{g}_s(z) u_s^* \in T_{z_s} \mathcal{Z}_{\alpha_s}. \quad (6.11)$$

Achieving the virtual constraints by zeroing the corresponding outputs reduces the dimension of the system by restricting its dynamics to the submanifold \mathcal{Z}_{α_s} embedded in the continuous-time state space TQ_s . \mathcal{Z}_{α_s} is called the zero dynamics manifold and the restriction dynamics $\dot{z}_s = f_s^*|_{\mathcal{Z}_{\alpha_s}}(z)$ is called the zero dynamics. As we will see next, the zero dynamics is actuated.

As was done in walking, from the Lagrangian dynamics, a valid set of coordinates on

Z_{α_s} is

$$x_{zd}^s = \begin{bmatrix} \xi_1^s \\ \xi_2^s \\ \xi_3^s \\ \xi_4^s \\ \xi_5^s \\ \xi_6^s \end{bmatrix} = \begin{bmatrix} \theta_s \\ q_{Bspst} \\ q_{mLSst} \\ \frac{\partial \mathcal{L}_s}{\partial \dot{q}_{Bspst}} \\ \frac{\partial \mathcal{L}_s}{\partial \dot{q}_{mLSst}} \\ \frac{\partial \mathcal{L}_s}{\partial \dot{q}_{Tor}} \end{bmatrix}. \quad (6.12)$$

This set of coordinates explicitly contains the B_{spring} variable, which illustrates clearly that the zero dynamics is compliant:

$$\dot{x}_{zd}^s = \begin{bmatrix} \dot{\xi}_1^s \\ \dot{\xi}_2^s \\ \dot{\xi}_3^s \\ \dot{\xi}_4^s \\ \dot{\xi}_5^s \\ \dot{\xi}_6^s \end{bmatrix} = \begin{bmatrix} \mathcal{L}_{\tilde{f}_s} \theta_s \\ \mathcal{L}_{\tilde{f}_s} q_{Bspst} \\ \mathcal{L}_{\tilde{f}_s} q_{mLSst} \\ \frac{\partial \mathcal{L}_s}{\partial q_{Bspst}} + \tau_{sp} \\ \frac{\partial \mathcal{L}_s}{\partial q_{mLSst}} \\ \frac{\partial \mathcal{L}_s}{\partial q_{Tor}} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ u_{mLSst} \\ 0 \end{bmatrix} \quad (6.13)$$

Since the stance motor leg shape input explicitly appears in the zero dynamics, in addition to being compliant, the zero dynamics for the stance phase is also actuated. The force control we intend to implement needs to be a function of the state on the zero dynamics, i.e., $u_{mLSst} = u_{mLSst}(z_s)$.

6.5 Active Force Control - Virtual Compliance

Through the choice of the virtual constraints, specifically choosing only three control variables on which the constraints are imposed, the stance motor leg shape input was left free and not used for imposing a virtual constraint. Thus we have a choice of the feedback control to impose on this input. Among all the different feedbacks that one can implement,

we choose something very simple - a virtual compliant element. By defining the feedback,

$$u_{\text{mLS}_{\text{st}}}(x_{\text{s}}) = -k_{\text{vc}}(q_{\text{mLS}_{\text{st}}} - q_{\text{mLS}_{\text{vc}}}), \quad (6.14)$$

a virtual compliant element of stiffness k_{vc} , and rest position $q_{\text{mLS}_{\text{vc}}}$ is implemented using the motor leg shape actuator. An additional damping element could be added if desired. The transmission of MABEL places this virtual compliant element in series with the physical compliance. Since both these compliances are in series, this method provides a means of dynamically varying the effective compliance of the system.

This method of creating a virtual compliant element using the choice of virtual constraints was instrumental in maintaining good ground contact forces for large step-down experiments of up to 5 inches (see [50].) In Section 6.12, we show several more uses of virtual compliance. With this method, we can easily account for cable stretch that was not part of the model for the control design. Further, as suggested in [101], depending on the knee angle at impact, the spring force appearing at the stance knee has a nonlinear relation to the spring compression. Using virtual compliance, one can vary the effective compliance based on the knee angle at impact to account for this phenomenon, thereby preventing the stance knee from excessively bending. Virtual compliance can also be used to easily account for asymmetry in the robot. Further, there is a reduction in the number of parameters to be found in the optimization problem which will be discussed in Section 6.9. Finally, another potential benefit would be to use this method for rapid motions, where moving a joint from one position to another as fast as possible is sought rather than accurately moving it along a desired trajectory between two points as typically done in virtual constraints.

One potential disadvantage would be that we are trying to implement compliance using an actuator. On its own, this has several problems associated with high bandwidth requirements on the actuator, and bad efficiency since the actuator would be required to do negative work. However, in this case, since the virtual compliance is in series with a physical compliance, the actual spring would handle the high bandwidth and potentially perform any negative work. In which case, this is not a severe disadvantage of the proposed method. Further analysis, beyond the scope of the current work, should be done to confirm

this.

For future use, we assemble the independent parameters of the virtual compliance as $\alpha_{\text{vc}} \in \mathbb{R}^2$ and defined as

$$\alpha_{\text{vc}} = \begin{bmatrix} k_{\text{vc}} \\ q_{\text{mLSvc}} \end{bmatrix}. \quad (6.15)$$

6.6 Virtual Constraint Design for Flight

During the flight phase, neither foot is in contact with the ground. The stance leg refers to the leg that was the stance leg in the previous stance phase, and similarly for the swing leg. For the flight phase, we have four actuators available to impose virtual constraints. One virtual constraint is designed per independent actuator.

The virtual constraints in the flight phase are parametrized by θ_{f} , a strictly monotonic function of the joint configuration variables. For running, we choose θ_{f} to be the horizontal position of the hip¹, i.e.,

$$\theta_{\text{f}}(q_{\text{f}}) = p_{\text{hip}}^h. \quad (6.16)$$

The virtual constraints for the flight phase can then be expressed in the form

$$y_{\text{f}} = h_{\text{f}}(q_{\text{f}}) = H_0^{\text{f}} q_{\text{f}} - h_d^{\text{f}}(\theta_{\text{f}}). \quad (6.17)$$

Next we have a choice of which variables we choose to be the control variables on which the virtual constraints are imposed. For the flight phase, on the stance leg, the leg angle and the motor leg shape are chosen. The stance foot needs to be lifted off from the ground rapidly and this can be achieved by bending the leg by repositioning the stance motor leg shape, and also by moving the leg backward by repositioning the stance leg angle. On the swing leg, the swing foot needs to be unfolded in preparation for an impact. This can be achieved by repositioning the swing motor leg shape. Finally, for directly specifying the touchdown angle, the absolute leg angle of the swing leg is taken as a control variable. This

¹Ideally, the horizontal position of the COM would be a good choice, since it is guaranteed to be strictly monotonic in the flight phase. However, for experimental convenience, and due the fact that the torso for MABEL is heavy and the legs relatively light weight, the horizontal position of the hip would also be monotonic.

has an added advantage that if the torso pitches forward excessively, the swing leg angle automatically repositions such that the absolute leg angle is the desired value at touchdown. In summary, the controlled variables are

$$H_0^f q_f = \begin{bmatrix} q_{mLS_{st}} \\ q_{LA_{sw}} + q_{Tor} \\ q_{mLS_{sw}} \\ q_{LA_{st}} \end{bmatrix}. \quad (6.18)$$

The virtual constraints are parametrized by 5th order Bézier polynomials. The desired evolution of each of the controlled variables are denoted by $h_{mLS_{st}}^{d,f}$, $h_{LA_{abssw}}^{d,f}$, $h_{mLS_{sw}}^{d,f}$, and $h_{LA_{st}}^{d,f}$ respectively with corresponding Bézier coefficients $\alpha_{mLS_{st}}^f$, $\alpha_{LA_{abssw}}^f$, $\alpha_{mLS_{sw}}^f$, and $\alpha_{LA_{st}}^f$. The desired evolution of the virtual constraints are assembled as

$$h_d^f(\theta_f, \alpha_f) = \begin{bmatrix} h_{mLS_{st}}^{d,f}(\theta_f, \alpha_f) \\ h_{LA_{abssw}}^{d,f}(\theta_f, \alpha_f) \\ h_{mLS_{sw}}^{d,f}(\theta_f, \alpha_f) \\ h_{LA_{st}}^{d,f}(\theta_f, \alpha_f) \end{bmatrix}, \quad (6.19)$$

where the Bézier coefficients are organized as

$$\alpha_f = \begin{bmatrix} \alpha_{mLS_{st}}^f \\ \alpha_{LA_{abssw}}^f \\ \alpha_{mLS_{sw}}^f \\ \alpha_{LA_{st}}^f \end{bmatrix}. \quad (6.20)$$

6.7 Flight Zero Dynamics

The flight zero dynamics is relatively straightforward since all actuators are employed to enforce virtual constraints. The output function y_f is associated with the continuous flight

dynamics defined in (3.33). Differentiating the output twice with respect to time results in

$$\frac{d^2 y_f}{dt^2} = L_{f_f}^2 h_f(x_f, \alpha_f) + L_{g_f} L_{f_f} h_f(q_f, \alpha_f) u, \quad (6.21)$$

where $L_{g_f} L_{f_f} h_f(q_f, \alpha_f)$, the decoupling matrix, has full rank. Under the conditions of [130, Lemma 5.1],

$$u_f^*(x_f, \alpha_f) := -(L_{g_f} L_{f_f} h_f(q_f, \alpha_f))^{-1} L_{f_f}^2 h_f(x_f, \alpha_f), \quad (6.22)$$

is the unique control input that renders the smooth six-dimensional embedded submanifold

$$\mathcal{Z}_{\alpha_f} = \{x_f \in TQ_f \mid h_f(q_f, \alpha_f) = 0, L_{f_f} h_f(x_f, \alpha_f) = 0\} \quad (6.23)$$

invariant under the flight dynamics (3.33); that is, for every $z_f \in \mathcal{Z}_{\alpha_f}$,

$$f_f^*(z_f) := f_f(f) + g_f(z_f) u_f^* \in T_{z_f} \mathcal{Z}_{\alpha_f}. \quad (6.24)$$

Achieving the virtual constraints by zeroing the corresponding outputs reduces the dimension of the system by restricting its dynamics to the submanifold \mathcal{Z}_{α_f} embedded in the continuous-time state space TQ_f . \mathcal{Z}_{α_f} is called the zero dynamics manifold and the restriction dynamics $\dot{z}_f = f_f^*|_{\mathcal{Z}_{\alpha_f}}(z_f)$ is called the zero dynamics.

From Lagrangian dynamics, a valid set of coordinates on \mathcal{Z}_{α_f} is

$$x_{zd}^f = \begin{bmatrix} \xi_1^f \\ \xi_2^f \\ \xi_3^f \\ \xi_4^f \\ \xi_5^f \\ \xi_6^f \end{bmatrix} = \begin{bmatrix} q_{\text{Tor}} \\ p_{\text{hip}}^h \\ p_{\text{hip}}^v \\ \frac{\partial \mathcal{L}_f}{\partial \dot{q}_{\text{Tor}}} \\ \frac{\partial \mathcal{L}_f}{\partial \dot{p}_{\text{hip}}^h} \\ \frac{\partial \mathcal{L}_f}{\partial \dot{p}_{\text{hip}}^v} \end{bmatrix}. \quad (6.25)$$

This choice of coordinates is different from those chosen for RABBIT in [130, Chap. 9].

Since the flight dynamics for MABEL, developed in Section 3.2.5, incorporates the boom

dynamics, the angular momentum is not strictly conserved and a different set of coordinates for the zero dynamics is required. The zero dynamics is then given by

$$\dot{x}_{zd}^f = \begin{bmatrix} \dot{\xi}_1^f \\ \dot{\xi}_2^f \\ \dot{\xi}_3^f \\ \dot{\xi}_4^f \\ \dot{\xi}_5^f \\ \dot{\xi}_6^f \end{bmatrix} = \begin{bmatrix} L_{f_f} q_{\text{Tor}} \\ L_{f_f} p_{\text{hip}}^h \\ L_{f_f} p_{\text{hip}}^v \\ \frac{\partial \mathcal{L}_f}{\partial q_{\text{Tor}}} \\ \frac{\partial \mathcal{L}_f}{\partial p_{\text{hip}}^h} \\ \frac{\partial \mathcal{L}_f}{\partial p_{\text{hip}}^v} \end{bmatrix}. \quad (6.26)$$

6.8 Event Transitions

The division of running into the stance and flight phases necessitates the specification of transition maps between the phases. In preparation for the next section, we introduce correction polynomials so as to obtain hybrid invariance of the zero dynamics manifolds. We also model the hybrid dynamics on the zero dynamics manifold by concatenating the solutions of the parameter dependent hybrid systems for each subphase.

On transition from one phase to another we require the solution to be on the zero dynamics of the next phase. This ensures the zero dynamics manifold is hybrid invariant and enables us to study the behavior of the restricted hybrid system. Hybrid invariance is achieved by introducing correction polynomials [84, 48] which are parametrized by Bézier coefficients and are updated at event transitions such that the post-transition state lies in the zero dynamics manifold of the next phase. This is obtained by modifying the virtual constraint at event transitions by introducing new outputs for each of the the phases $p \in \mathcal{P}$

$$\begin{aligned} y_c^p &= h_p(q_p, \alpha_p, \alpha_c^p) \\ &= H_0^p q_p - h_d^p(\theta_p, \alpha_p) - h_c^p(\theta_p, \alpha_c^p). \end{aligned} \quad (6.27)$$

The output consists of the previous output (6.2), or (6.17), and an additional correction term h_c^p such that the post transition output and its velocity are zero, i.e., $y_c^{p+} = 0, \dot{y}_c^{p+} = 0$. This is achieved by choosing the Bézier coefficients α_c^p appropriately. Moreover, the correction

polynomial is designed in such a way that the modified virtual constraint is smoothly joined to the original virtual constraint at the middle of the current phase. The zero dynamics defined in Sections 6.4, and 6.7 can be defined to incorporate the new output with correction polynomial to obtain the zero dynamics manifolds $\mathcal{Z}_{\alpha_p, \alpha_c^p}$.

Next, the stance phase of running is broken into two subphases: the stance-compression and the stance-decompression. The division of the stance phase is artificial with respect to the virtual constraints, since for the three virtual constraints, the two subphases are combined into a single phase. The purpose of the subphases is to implement a deadbeat event-based control strategy to only change the parameters of the virtual compliance. The hybrid zero dynamics model for running that captures the continuous-time dynamics of the system in stance-compression subphase, stance-decompression subphase, the flight phase, and the discrete transitions among them, is given by

$$\Sigma_{zd}^s : \left\{ \begin{array}{l} \Sigma_{zd}^{sc} : \left\{ \begin{array}{l} z_s \in \mathcal{Z}_{\alpha_s, \alpha_c^s}, \quad \alpha_{vc} \in \mathbb{R}^2 \\ \dot{z}_s = f_s^*(z_s), \quad \dot{\alpha}_{vc} = 0 \\ \mathcal{S}_{sc \rightarrow sd} = \{z_s \in \mathcal{Z}_{\alpha_s, \alpha_c^s} \mid H_{sc \rightarrow sd}(z_s) = 0\} \\ z_s^{sd+} = \Delta_{sc \rightarrow sd}(z_s^{sc-}), \quad \alpha_{vc}^{sd+} = \alpha_{vc}^{sd}, \end{array} \right. \\ \Sigma_{zd}^{sd} : \left\{ \begin{array}{l} z_s \in \mathcal{Z}_{\alpha_s, \alpha_c^{sd}}, \quad \alpha_{vc} \in \mathbb{R}^2 \\ \dot{z}_s = f_s^*(z_s), \quad \dot{\alpha}_{vc} = 0 \\ \mathcal{S}_{sd \rightarrow f} = \{z_s \in \mathcal{Z}_{\alpha_s, \alpha_c^s} \mid H_{sd \rightarrow f}(z_s) = 0\} \\ z_f^+ = \Delta_{sd \rightarrow f}(z_s^{sd-}), \end{array} \right. \end{array} \right.$$

$$\Sigma_{zd}^f : \left\{ \begin{array}{l} z_f \in \mathcal{Z}_{\alpha_f, \alpha_c^f} \\ \dot{z}_f = f_f^*(z_f) \\ \mathcal{S}_{f \rightarrow sc} = \{z_f \in \mathcal{Z}_{\alpha_f, \alpha_c^f} \mid H_{f \rightarrow sc}(z_f) = 0\} \\ z_s^+ = \Delta_{f \rightarrow sc}(z_f^-), \quad \alpha_{vc}^{sc+} = \alpha_{vc}^{sc}, \end{array} \right.$$

where α_{vc}^p for $p \in \mathcal{P}_s := \{sc, sd\}$ are independent parameters for the virtual compliance and are to be specified.

The switching surfaces for the transitions are defined by the zero level sets of the threshold functions given below

$$\begin{aligned}
H_{sc \rightarrow sd} &:= \theta_s - \theta_{sd} \\
H_{sd \rightarrow f} &:= H_{s \rightarrow f} \\
H_{f \rightarrow sc} &:= H_{f \rightarrow s},
\end{aligned} \tag{6.28}$$

where θ_{sd} is a independent parameter to be specified, $H_{s \rightarrow f}$ and $H_{f \rightarrow s}$ are as as defined in Sections 3.2.6, 3.2.7 respectively. Note that the stance to flight threshold function is defined as $H_{s \rightarrow f} : \mathcal{S}_{s \rightarrow f} \times \mathcal{U} \rightarrow \mathbb{R}$. However, with the inputs all being a function of the state (refer (6.9), (6.14)), this transition is then essentially defined on the stance to flight switching surface, $\mathcal{S}_{s \rightarrow f}$. The transition maps provide the initial conditions for the ensuing phase and are given below

$$\begin{aligned}
\Delta_{sc \rightarrow sd} &:= id \\
\Delta_{sd \rightarrow f} &:= \Delta_{s \rightarrow f} \\
\Delta_{f \rightarrow s} &:= \Delta_{f \rightarrow s},
\end{aligned} \tag{6.29}$$

with $\Delta_{s \rightarrow f}$, $\Delta_{f \rightarrow s}$ are as defined in Sections 3.2.6, 3.2.7 respectively. Note that the stance to flight transition function is defined as $\Delta_{s \rightarrow f} : TQ_s \times \mathcal{U} \rightarrow TQ_f$. However, as mentioned earlier, since the inputs are a function of the state, this mapping is essentially from the stance state space.

To find the set of values for the independent parameters of the constraint design, and parameters of the virtual compliance, we employ the above restricted lower-dimensional hybrid system and formulate the problem as a constrained optimization.

6.9 Gait Design Through Optimization

A periodic running gait is designed by selecting the free parameters in the virtual constraints, and the virtual compliance. As was carried out for gait design for walking in Section 4.4, an optimization problem is posed to minimize energy per step length, subject to constraints to meet periodicity, workspace and actuator limitations. The equations of the hybrid zero dynamic model developed in the earlier section, which are of reduced dimension compared to the full dynamics, are employed for efficiency of computation.

The cost function

$$J_{nom}(\alpha_s, \alpha_f, \alpha_{vc}^{sc}, \alpha_{vc}^{sd}) = \frac{1}{p_{toesw}^h (q_f^-)} \int_0^{T_I} \|u(t)\|^2 dt, \quad (6.30)$$

where T_I is the step duration (stance plus flight time) and p_{toesw}^h is the step length. Minimizing this cost function tends to reduce peak torque demands and minimize the electrical energy consumed per step.

6.10 Fixed Point for Running

This section presents a nominal fixed point of 1.34 m/s obtained by applying the optimization procedure outlined in Section 6.9 to the virtual constraints of Sections 6.3, 6.6, the virtual compliance of Section 6.5, and with the cost function (6.30). Figure 6.2 illustrates the nominal evolution of the virtual constraints for the stance flight phases along with other configuration variables for one step of running. The squares on the plots indicate the transition from stance to flight phase. The step time is 525 ms with 69% spent in stance and 31% in flight. On entry into the flight phase, the torso is leaning forward (negative torso angle) and is rotating backward (positive torso velocity). The swing leg angle travels roughly 57% of its total 47.5° during the stance phase² and needs to travel the remaining 43% in the flight phase which is of smaller duration. Thus the velocities of the joints in the flight are high compared to those of the stance phase. The instantaneous change in the stance motor leg shape position on transition to flight is to reset the stance spring to its rest position in the flight phase.

Figure 6.3 illustrates the evolution of the leg shape and the stance B_{spring} variables. The squares in the plot indicate the stance to flight transition and the circle in the spring plot indicates the stance-compression to stance-decompression transition. During the flight phase, the stance leg shape initially unfolds due to the large velocity of push-off during the final part of the stance phase as the spring rapidly decompresses. During the stance-compression phase the spring compresses, reaches its peak value of almost 36° and starts to decompress.

²Contrast this to that of humans, where the legs travel roughly 90% of the range of travel during the stance phase.

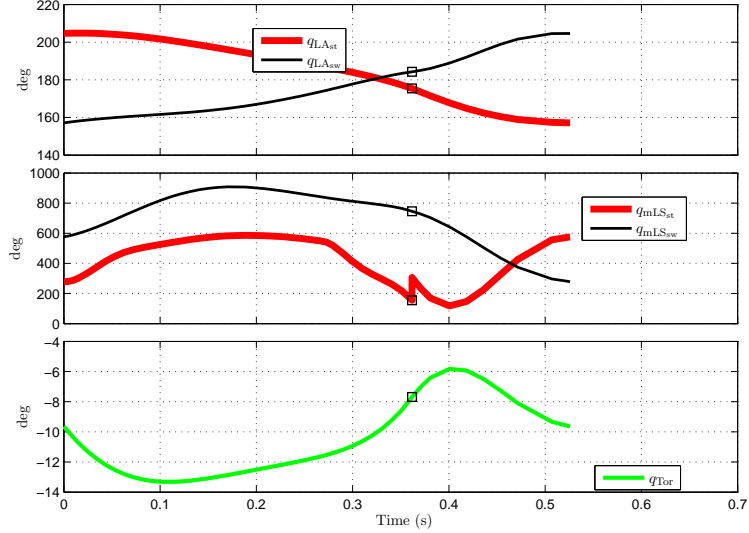


Figure 6.2: Evolution of the virtual constraints and configuration variables for a nominal fixed point (periodic running gait) at a speed of 1.34 m/s and step length 0.7055 m. The squares illustrate the location of transition between stance to flight phase.

On transition to the stance-decompression phase, the motor injects energy into the system causing the spring to rapidly compress to a peak of 47° . At lift-off, when the vertical component of the ground force goes to zero, the spring is compressed to approximately 25° .

Figure 6.4 illustrates the actuator torques used to realize the gait. The stance and swing leg angle torques and the swing leg shape torque are small compared to the peak torque capacities of the actuators: 30Nm. The stance leg shape torque is large, initially to support the weight of the robot as the stance knee bends and subsequently to provide a large energy injection in the stance-decompression phase to achieve lift-off. The stance motor leg shape torque is discontinuous at the stance-compression to stance-decompression transition due to an instantaneous change in the parameters for the virtual compliance. All torques are discontinuous on the stance to flight transition due to the impact of the spring with the hard-stop.

Figure 6.5 illustrates the evolution of the swing leg height and the vertical position of the center of mass of the robot. The swing foot is over 15 cm above the ground at its peak to offer good ground clearance for hard impacts. During the stance phase, the COM undergoes an asymmetric motion with the lowest point of potential energy being around

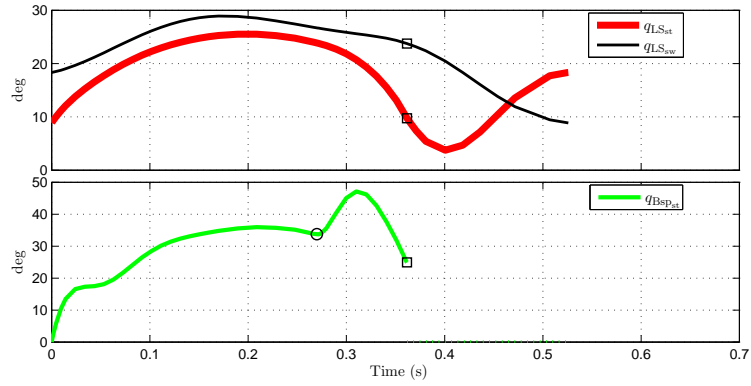


Figure 6.3: Evolution of the leg shape and stance B_{spring} variables corresponding to the nominal fixed point. The squares illustrate the location of transition between stance to flight phase. The circle on the B_{spring} plot illustrates the location of the sc to sd event transition.

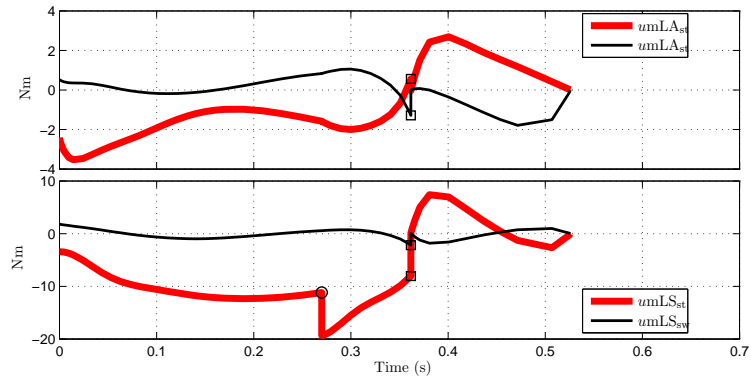


Figure 6.4: Actuator torques corresponding to the nominal fixed point. The squares illustrate the location of transition between stance to flight phase. The circle on the $umLS_{st}$ plot illustrates the location of the sc to sd event transition. Note that the torques are discontinuous at stance to flight transitions. Also note the additional discontinuity for $umLS_{st}$ at the sc to sd event transition due to the instantaneous change in the offset for the virtual compliance at this transition.

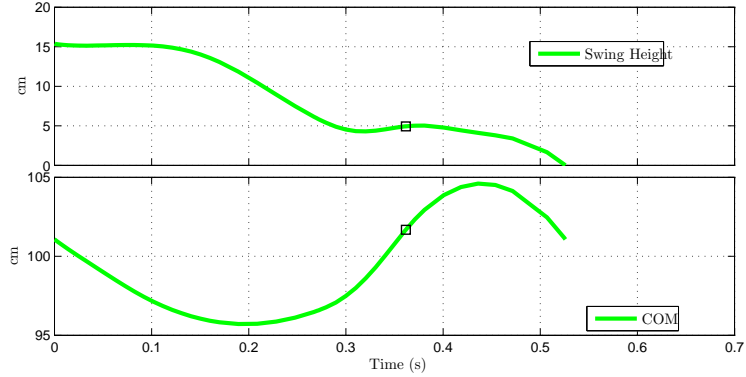


Figure 6.5: Evolution of swing leg height and vertical center of mass (COM) of the robot for the nominal fixed point. The COM trajectory clearly illustrates the lowest point of potential energy during the stance phase and the ballistic trajectory in the flight phase, both of which are dominating characteristics of running. The squares illustrate the location of transition between stance to flight phase.

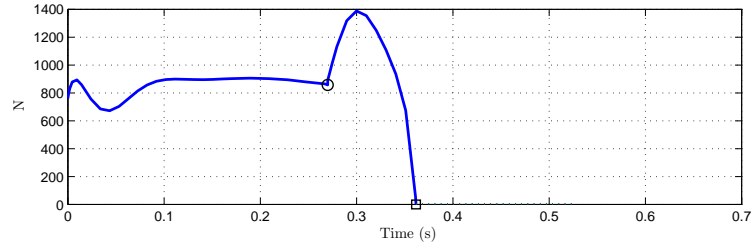


Figure 6.6: Vertical component of the ground reaction force for the nominal running fixed point. At the sc to sd event transition (indicated by the circle), the change in the offset for the virtual compliance causes the spring to compress further which increases the ground reaction force considerably. Takeoff occurs when the ground reaction force goes to zero (indicated by the square.)

52% into the stance phase. During the flight phase, the COM has a ballistic trajectory. Both these motions are dominant characteristics of running. Figure 6.6 illustrates the vertical component of the ground reaction force. Immediately upon impact, during the stance-compression phase, there is a peak in the ground reaction force due to the spring compressing rapidly on impact. During most of the stance-compression phase, the force is fairly constant. On transition to stance-decompression phase, the energy injection causes the force to rapidly first increase and then go to zero at which point stance to flight transition occurs.

6.11 Closed-loop Design and Stability Analysis

The feedback presented in (6.9), (6.14), (6.22), when used with the modified outputs (6.27), renders the zero dynamics hybrid invariant. This feedback does not however render the solution stable or attractive in any way. In the following, we introduce control action on three levels with an inner-loop and two outer-loops. On the first level, a continuous-time controller is presented that in addition to rendering the zero dynamics invariant also makes it attractive. The hybrid invariance is still achieved through the correction polynomials on a event to event level. On the second level, an outer-loop event-based discrete linear controller is introduced to exponentially stabilize the periodic orbit representing the running gait. Finally on the third level, an additional outer-loop event-based discrete nonlinear controller is introduced to enlarge the domain of attraction of the periodic orbit.

The classic input-output linearizing controller

$$u = u^*(x_p, \alpha_p) - L_{g_p} L_{f_p} h_p(q_p, \alpha_p)^{-1} \left(\frac{K_{p,P}}{\epsilon^2} y_c^p + \frac{K_{p,D}}{\epsilon} \dot{y}_c^p \right), \quad (6.31)$$

where $p \in \mathcal{P}$, renders the zero dynamics both invariant and attractive. The correction polynomials create hybrid invariance. For ϵ sufficiently small, the stability of the fixed point under this control action can be analyzed through use of the restricted Poincaré map [82], i.e., the Poincaré map associated with the invariant hybrid system presented in Section 6.8. We consider $\mathcal{S}_{sc \rightarrow sd}$ as a Poincaré section. Then, the stability of the fixed point can be determined by the restricted Poincaré map defined as $\rho : \mathcal{S}_{sc \rightarrow sd} \cap \mathcal{Z}_{\alpha_s, \alpha_c^s} \rightarrow \mathcal{S}_{sc \rightarrow sd} \cap \mathcal{Z}_{\alpha_s, \alpha_c^s}$. Using this restricted Poincaré map, we can numerically calculate the eigenvalues of its linearization about the fixed point. The analysis shows that the running gait obtained by optimizing (6.30) and with the closed-loop controller (6.31) is unstable³ with a dominant eigenvalue of 1.1928. Thus, an additional controller needs to be designed to stabilize the running fixed point.

³In fact all running fixed points that were found were unstable.

6.11.1 Exponentially Stabilizing Outer-loop Controller

The dominant eigenvalue of the linearization of the restricted Poincaré map about the fixed point indicates that the fixed point is unstable. An outer-loop discrete event-based linear controller can be designed to stabilize the discrete linear system representing the linearized Poincaré map, as was done for Thumper in [91]. We identify certain parameters that can be varied step-to-step, and which could possibly affect stability of the fixed point. These are assembled as $\beta \in \mathcal{B}$

$$\beta = \begin{bmatrix} \beta_{k_{vc}}^{sc} \\ \beta_{k_{vc}}^{sd} \\ \beta_{TD} \\ \beta_{q_{mLSvc}}^{sc} \\ \beta_{q_{mLSvc}}^{sd} \\ \beta_{\theta_f} \\ \beta_{Tor} \end{bmatrix}, \quad (6.32)$$

where $\beta_{k_{vc}}^p, \beta_{q_{mLSvc}}^p$ are the virtual compliance stiffness and offset for phase p , β_{TD} is the touchdown angle, β_{θ_f} is an offset to be added to θ_f^- , and β_{Tor} is the torso offset.

Next, since determining the state z_s on the zero dynamics manifold in an experimental implementation is subject to measurement noise, the full-order Poincaré map is considered for the design of the outer-loop event-based controller. This map is defined as $P : \mathcal{S}_{sc \rightarrow sd} \times \mathcal{B} \rightarrow \mathcal{S}_{sc \rightarrow sd}$, such that

$$x_s^{sc-}[k+1] = P(x_s^{sc-}[k], \beta[k]). \quad (6.33)$$

The full-order Poincaré map is linearized about the fixed point to obtain the discrete-time linear system

$$\delta x_s^{sc-}[k+1] = \left. \frac{\partial P}{\partial x_s^{sc-}} \right|_{(x_s^{sc-*}, 0)} \delta x_s^{sc-}[k] + \left. \frac{\partial P}{\partial \beta} \right|_{(x_s^{sc-*}, 0)} \beta[k], \quad (6.34)$$

where $\delta x_s^{sc-} = x_s^{sc-} - x_s^{sc-*}$. Discrete LQR is used to design a linear feedback such that the

closed-loop eigenvalues are within the unit circle. The feedback can be written as

$$\beta[k] = \Gamma^\beta(x_s^{sc-}) := K_{LQR}(x_s^{sc-} - x_s^{sc-*}). \quad (6.35)$$

This procedure is carried out numerically, and for the presented fixed point, the dominant eigenvalue of the Poincaré map with the feedback Γ^β is found to be 0.8383, which concludes that the fixed point is locally exponentially stabilized with this controller.

Next, in preparation for experimental validation, we study the robustness of the controller to perturbations. From the walking experiments, we observe that successful gaits were when the controller could reject perturbations in the form of external forces (see Section 5.1.2), ground variations and structural modifications (see [45]). For running, with the feedback controller comprised of the continuous-time control Γ_p^α , the discrete event-based controller $\Gamma_p^{\alpha_c}$ as the inner-loop, and with the discrete event-based controller Γ^β as the outer-loop, the robustness to external perturbations is studied. This controller can reject an error in torso of up to 6° in both directions. This is fairly good robustness to perturbations in torso angle. However, as shown in Figure 6.7(a), the controller is unable to reject an error in the form of the stance leg shape being bent by 5° . Thus, there is a need for a controller that can improve the domain of attraction of the fixed point. This will be crucial for experimental validation.

6.11.2 Domain of Attraction Enlarging Outer-loop Nonlinear Controller

Although the outer-loop controller, Γ^β , exponentially stabilizes the running gait, obtaining a larger domain of attraction is not achieved with the LQR method. This section details a discrete-time event-based control design that acts on the hybrid closed-loop system with the controllers Γ_μ^α , $\Gamma_p^{\alpha_c}$, and Γ^β .

The closed-loop system with the continuous-time and discrete-time controllers Γ_μ^α , $\Gamma_p^{\alpha_c}$ forming the inner loop to create hybrid invariant and attractive surfaces in the stance and flight state spaces, and the outer-loop controller Γ^β exponentially stabilizing the periodic orbit is considered. The closed-loop hybrid system can be once again sampled using the method of Poincaré and an event-based controller can be designed. We consider the switch-

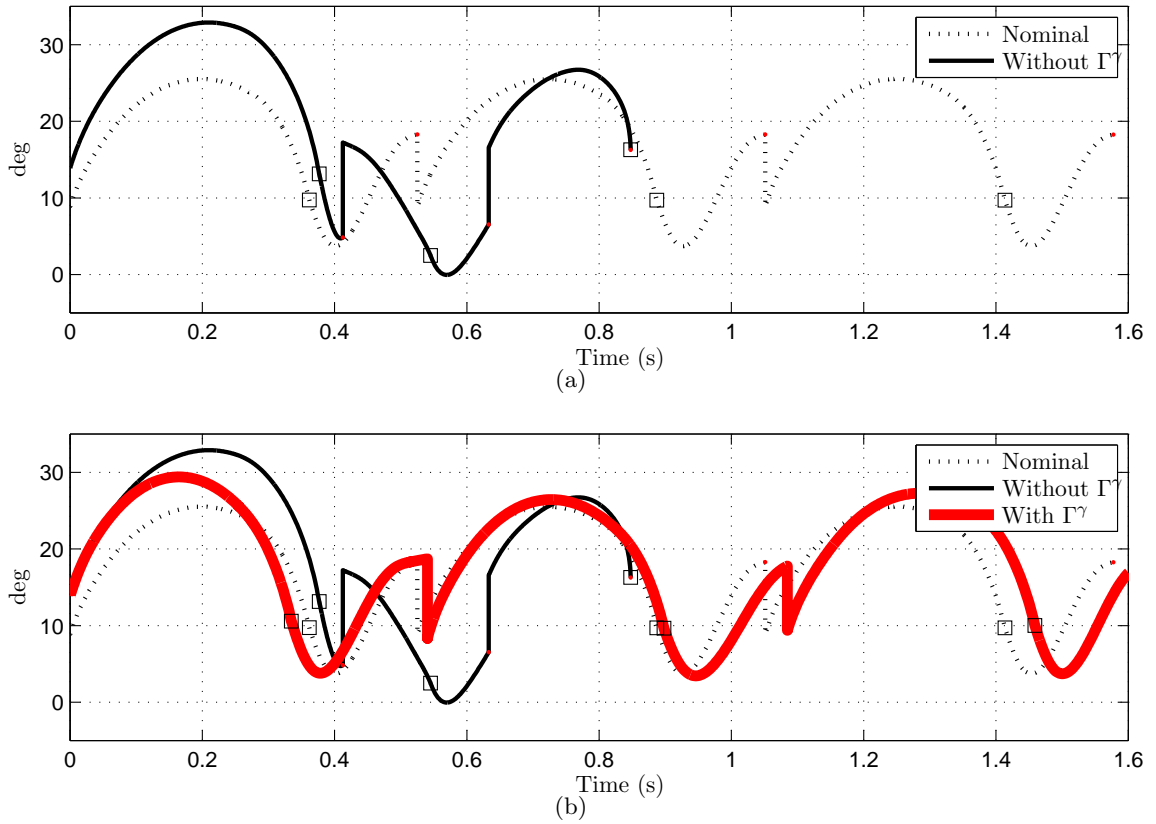


Figure 6.7: Three step simulation of a 5° perturbation in the impact value of the leg shape. (a) With out Γ^γ outer-loop controller, (b) With Γ^γ outer-loop controller. The squares and dots on the plots indicate locations at which the controller transitions to from stance to flight phase, and flight to stance phase, respectively.

ing surface $\mathcal{D} := \Delta_{f \rightarrow s}(\mathcal{S}_{f \rightarrow s}) \subset TQ_s$ as the Poincaré section. A set of parameters $\gamma \in \mathcal{G}$ is considered and the Poincaré map $P : \mathcal{D} \times \mathcal{G} \rightarrow \mathcal{D}$ is studied. The γ -parameters are assembled as

$$\gamma = \begin{bmatrix} \gamma_{k_{vc}}^{sc} \\ \gamma_{\text{Tor}} \\ \gamma_{\text{LS}_{sw}} \\ \gamma_{\delta_{sc \rightarrow sd}} \end{bmatrix}, \quad (6.36)$$

where $\gamma_{k_{vc}}^{sc}$ is the virtual compliance stiffness for the stance-compression subphase, γ_{Tor} is the torso offset, $\gamma_{\text{LS}_{sw}}$ is an offset that lifts the swing leg higher, $\gamma_{\delta_{sc \rightarrow sd}}$ is an offset that shifts the stance-compression to stance-decompression transition event.

Further, to keep track of the previous state on the Poincaré section, we define the discrete map $\mathcal{F} : \mathcal{D} \times \mathcal{D} \times \mathcal{G} \rightarrow \mathcal{D} \times \mathcal{D}$ as

$$\begin{bmatrix} x_s^+[k+1] \\ x_s^{+prev}[k+1] \end{bmatrix} = \mathcal{F}(x_s^+[k], x_s^{+prev}[k], \gamma[k]) := \begin{bmatrix} P(x_s^+[k], \gamma[k]) \\ x_s^+[k] \end{bmatrix}. \quad (6.37)$$

A nonlinear controller is designed to modify the γ -parameters based on state on the switching surface. The control design is motivated by insight into controlling simpler hopping models such as the SLIP. Essentially, the following control actions are performed by the nonlinear event-based controller, Γ^γ . Based on the speed difference from the current step and the fixed point value, γ_{Tor} is updated to lean the torso to reduce the error in speed. Further, with faster speeds, the energy injected during the stance-decompression phase reduces since the time spent in this phase decreases with faster speeds. To account for this, the position of the stance-compression to stance-decompression transition is changed, by updating $\gamma_{\delta_{sc \rightarrow sd}}$, to effectively increase or decrease the time spent in the stance-decompression phase injecting energy. Next, the knee angle at impact is relatively important. As suggested by Rummel and Seyfarth in [101], for segmented legs with compliance in the joints, larger leg compression is required to produce the same leg force when the rest position of the springs are moved away from that corresponding to straight legs. Since, for MABEL, the rest position of spring is essentially the impact angle of the knee (the spring is at its rest position

until there are large forces on the toe,) for a more bent knee at impact, the leg would have to collapse more to produce spring forces to support the weight of the robot. To account for this, $\gamma_{k_{vc}}^{sc}$ changes the virtual compliance based on the impact angle of the knee. Further in anticipation of a more bent knee during stance, the swing leg shape is bent further through $\gamma_{LS_{sw}}$ for good ground clearance.

The control law Γ^γ is summarized as

$$\begin{aligned}
\gamma_{k_{vc}}^{sc} &= \begin{cases} K_{k_{vc}}^1 (q_{LS_{st}}^{s+} - q_{LS_{st}}^{s+*}), & q_{LS_{st}}^{s+} - q_{LS_{st}}^{s+*} > 0 \\ K_{k_{vc}}^2 (q_{LS_{st}}^{s+prev} - q_{LS_{st}}^{s+*}), & q_{LS_{st}}^{s+} - q_{LS_{st}}^{s+*} \leq 0, \quad q_{LS_{st}}^{s+prev} - q_{LS_{st}}^{s+*} > 0 \\ 0, & \textit{otherwise} \end{cases} \\
\gamma_{Tor} &= \begin{cases} K_{Tor}^1 (sp(x_s^+, x_s^{+prev}) - sp(x_s^{+*})), & (speed(x_s^+, x_s^{+prev}) - sp(x_s^{+*})) > 0 \\ K_{Tor}^2 (sp(x_s^+, x_s^{+prev}) - sp(x_s^{+*})), & (speed(x_s^+, x_s^{+prev}) - sp(x_s^{+*})) \leq 0 \\ 0, & \textit{otherwise} \end{cases} \\
\gamma_{LS_{sw}} &= \begin{cases} K_{LS_{sw}}^1 (q_{LS_{st}}^{s+} - q_{LS_{st}}^{s+*}), & q_{LS_{st}}^{s+} - q_{LS_{st}}^{s+*} > 0 \\ K_{LS_{sw}}^2 (q_{LS_{st}}^{s+prev} - q_{LS_{st}}^{s+*}), & q_{LS_{st}}^{s+} - q_{LS_{st}}^{s+*} \leq 0, \quad q_{LS_{st}}^{s+prev} - q_{LS_{st}}^{s+*} > 0 \\ 0, & \textit{otherwise} \end{cases} \\
\gamma_{\delta_{sc \rightarrow sd}} &= \begin{cases} K_{\delta_{sc \rightarrow sd}} (sp(x_s^+, x_s^{+prev}) - sp(x_s^{+*})), & (speed(x_s^+, x_s^{+prev}) - sp(x_s^{+*})) > 0 \\ 0, & \textit{otherwise} \end{cases}
\end{aligned} \tag{6.38}$$

where $(sp(x_s^+, x_s^{+prev}))$ is the average speed of the past step, and $sp(x_s^{+*})$ is the average speed of the fixed point. The gains $K_{k_{vc}}^i, K_{k_{vc}}^{i,sc}, K_{LS_{sw}}^i, K_{\delta_{sc \rightarrow sd}}$, with $i \in \{1, 2\}$ are iteratively found through simulations. Furthermore, the γ -parameters are bounded such that, $-\gamma^{sat} \leq \gamma \leq \gamma^{sat}$.

With this second outer-loop discrete event-based controller Γ^γ , the robustness to perturbations is increased and as shown in Figure 6.7(b), is able to reject a perturbation of 5° in the impact leg shape angle. It must be noted that the design of the controllers $\Gamma^\alpha, \Gamma^{\alpha_c}, \Gamma^\beta$ are carried out through rigorous mathematical techniques, whereas the design of the outer-loop controller Γ^γ is rather based on heuristics. Section 8.2 provides additional comments

in this regard.

6.12 Modifications for Experimental Implementation

The feedback controller design for running presented in the previous sections introduces control in three levels. On the first level, continuous-time feedback controllers Γ_p^α with $p \in \mathcal{P} := \{s, f\}$ are employed in the stance and flight phases to create invariant and attractive surfaces, $\mathcal{Z}_{(\alpha_p, \alpha_c^p, \beta, \gamma)} \subset TQ_p$, embedded in the state space for each of the respective phases. The discrete-time feedback controllers $\Gamma_p^{\alpha_c}$ are employed in the transitions between the phases which render these surfaces hybrid invariant. On the second level, an event-based controller Γ^β performs step to step changes to render the periodic orbit representing running embedded in these surfaces exponentially stable. Finally on the third level, the event-based controller Γ^γ performs step to step changes to increase the domain of attraction of the periodic orbit. However, this controller on its own will not work in experiments for reasons that are outlined next along with the required modifications to the controller.

All control design in this thesis is based on the model developed in Section 3.2, henceforth called the *simple model*. However, there are certain discrepancies between this model and the actual robot, that were only discovered through experiments. For the walking experiments reported earlier in Section 5.1, these discrepancies were relatively minor and did not adversely effect the experiments. However, based on hopping experiments reported in [50], these discrepancies are critical for running and need to be accounted for. A new model, the *complex model*, was developed to account for these discrepancies. A brief overview of the differences between the simple and complex models is presented in Appendix C, with the complete complex model presented in [50]. The simple model is utilized for design, while the complex model is utilized for validation. Performing optimization directly on the complex model is not computationally feasible.

The most critical of these model discrepancies is the cable stretch. The model presented in Section 3.2 assumes the transmission is rigid, i.e., the cables do not stretch. However in experiments, the cables stretches significantly. For running, cable stretch in the leg shape joint reaches a peak of almost 30° of knee motion during the stance phase, just prior to

lift-off. Seen in another way, almost 80% of motion in the knee is due to cable stretch. A controller that can account for this severe cable stretch is required.

The cable stretch in the leg shape joint can be modeled as a spring (with damping) and placed in series with the virtual compliant leg. This causes the compliance due to cable stretch to appear in series with the physical compliance and the motor leg shape actuator. Since the controller presented in the previous sections utilized active force control to create a virtual compliant element, all three sources of compliance occur in series. The optimization process produced a particular spring stiffness for the virtual compliance. Using the active force control, the virtual compliance can be changed such that the compliance due to the cable stretch and the virtual compliance in series together has the effective compliance of that designed by the optimization process. With this modification, the effective compliance of the leg is now the same as that without cable stretch, i.e., cable stretch has been accounted for by the control design.

With these modifications, the running controller is ready for experimental deployment.

Remark 6.1. Other discrepancies in the model can be attacked in a similar manner. For instance, due to the boom, there is an asymmetry in the robot - the robot weighs 10% heavier on the inner leg than on the outer leg. It is hypothesized that by making the effective compliance of the inner leg to be 10% stiffer by changing the virtual compliance on the inner leg, the asymmetry could be accounted for.

CHAPTER VII

Running Experiments

This chapter documents experimental implementations of the running controller developed in Chapter VI in various running scenarios. To illustrate the power and limitations of the proposed method, two experiments are presented. The first experiment details a running experiment with passive feet, while the second experiment details a running experiment with point feet.

Early running experiments executed on MABEL (without any hardware modifications) resulted in 10 or more steps of running before the experiment was terminated due to various reasons; initially, stable running (that is, an experiment resulting in potentially unbounded number of steps) was not obtained. Most experiments terminated due to one or more of the following reasons: Feet slipping; severe cable stretch in the leg angle cables; and inability of the controller to regulate speed, leading to high speed running and eventual failure.

In an attempt to address slipping and possibly speed regulation, hardware modifications were carried out on MABEL to replace the shins on both legs with a different pair of shins that terminate in passive feet¹ enclosed in a pair of regular running sneakers. The large surface area of the footprint could potentially prevent slipping and possibly slow down the robot. The running controller developed in the previous chapter was implemented on this system and stable running was achieved. An experiment where MABEL took one hundred running steps is examined in detail.

Once stable running with passive feet was obtained, the insight gained from the experiment was used to design and implement a successful running controller on MABEL with

¹The feet were designed by a team led by Hae-Won Park as part of a course project for ME 450.

point feet. An experiment where MABEL took one hundred and thirteen running steps is examined in detail.

In both experiments, the controller comprised of the inner-loop continuous-time compliant HZD controller with discrete-time event-based outer-loop controllers developed in Chapter VI is executed. A few minor differences exist between the two implementations and the controller design presented in the previous chapter. These differences are in the transition conditions between the various phases and the outer-loop event-based controllers, and are documented in detail in the following sections.

7.1 Experiments

As in the walking experiments reported in Chapter V, the controller was first coded in C++ and evaluated on a detailed simulation model of the robot that included encoder quantization and numerical estimation of velocity variables from encoder measurements. The controller was tested under various model perturbations, such as errors in the torso mass, spring stiffness, torso center of mass position, and deviations in initial conditions. The simulation model was then replaced with the physical robot.

The results of the experiments are presented in Figures 7.1-7.12. In the experiments, the left leg refers to the inner leg, which is closer to the center boom, and the right leg refers to the outer leg, which is farther from the center boom. All running speeds are measured with respect to the center point of the hip between the two legs. Videos of the experiments are available on YouTube [45].

7.1.1 Running with Passive Feet

Early running experiments always terminated with a maximum of 10 – 15 running steps. The experiments failed due to various reasons, either due to slipping, inability to regulate speed, cable stretch, and other hardware issues. To address the issue of slipping and potentially that of speed regulation, the following hardware modifications were made on MABEL. The shins terminating with point feet were replaced with shins terminating in passive feet. The passive feet were enclosed within regular running shoes. The new shins

with the feet and shoes are twice the weight of the old shins terminating in point feet, and being placed far from the hip, significantly change the inertia of the legs.

The passive feet with shoes address the problem of slipping. To understand better the problem of speed regulation, we take a look at the biomechanics literature. As emphasized by Cavagna et al. [15], in running, changes of forward kinetic energy and gravitational potential energy are in phase and therefore cannot exchange with one another to smooth out fluctuations, i.e., an increase in kinetic energy cannot be rejected by a corresponding decrease in potential energy. During the stance-decompression phase, the developed running controller performs energy injection to primarily increase the potential energy. However, as discussed, trying to reject an increase in speed by reducing this energy injection in the stance-decompression phase is not possible. In fact, this was observed in earlier experiments. At faster speeds, the time spent in the stance-decompression phase decreases, thereby reducing the effective energy injection. However, a speed reduction was not observed. Instead, the reduction in the energy injection led to a lower apex height in the flight phase. This led to adding a component to the outer-loop Γ^γ controller that modifies the location of the stance-compression to stance-decompression phase so as to maintain a constant energy injection even at faster speeds. Further, as suggested in [77, 79], animals vary stance leg stiffness to regulate running speed. Thus, there is a need for the controller to change the effective leg compliance as a function of speed.

To address this and other issues, the following additional modifications to the running controller presented in Chapter VI are performed.

Modifications to the event-based controllers $\Gamma^\beta, \Gamma^\gamma$: As just discussed, there is a need to vary the effective leg compliance as a function of speed. To do this, an additional component is introduced to the Γ^γ outer-loop event-based controller, which performs an event-based update of the virtual compliance in response to the error in speed,

$$\gamma_{k_{vc}}^{sc,add} = \begin{cases} -K_{k_{vc}} \sqrt{(sp(x_s^+, x_s^{+prev}) - sp(x_s^{+*}))}, & (speed(x_s^+, x_s^{+prev}) - sp(x_s^{+*})) > 0 \\ 0, & otherwise \end{cases} \quad (7.1)$$

Here, $sp(x_s^{+*})$ is the designed speed of the periodic running motion, and $sp(x_s^+, x_s^{+prev})$ is

the speed of the last step. $\gamma_{k_{vc}}^{sc,add}$ is then added to $\gamma_{k_{vc}}^{sc}$ computed in (6.38).

Next, to prevent very large values for the outer-loop event-based controller, Γ^β , the β -parameters are bounded such that, $-\beta^{sat} \leq \beta \leq \beta^{sat}$. At slow speeds, a large touchdown angle could potentially cause the foot to slip, and would require the β parameter corresponding to touchdown, β_{TD} , to be conservatively saturated. At fast speeds, a conservative saturation on β_{TD} limits how much the robot can slow down. To address this, the saturation for β_{TD} is changed as a function of the speed and is given by,

$$\beta_{TD}^{sat} = \begin{cases} 2^\circ, & 0 \leq sp(x_s^+, x_s^{+prev}) < 1.2 \\ 1.5^\circ, & 1.2 \leq sp(x_s^+, x_s^{+prev}) < 1.7 \\ 2^\circ, & 1.7 \leq sp(x_s^+, x_s^{+prev}) < 2 \\ 2.5^\circ, & 2 \leq sp(x_s^+, x_s^{+prev}) < 2.5 \\ 4^\circ, & 2.5 \leq sp(x_s^+, x_s^{+prev}) \end{cases} \quad (7.2)$$

Modifications to virtual compliance: The impacts of the feet equipped with shoes were softer than the impacts of the point feet. This resulted in the designed virtual spring constant k_{vc}^* being too stiff for running with feet. Consequently, the springs were not compressing sufficiently due to the softer impacts, causing the springs to decompress faster, thereby resulting in a flight phase with short duration and low ground clearance. To account for the softer impacts, the nominal virtual spring constant was reduced by 18% of the designed value.

Modifications to the switching surface $\mathcal{S}_{sc \rightarrow sd}$: On transition from the stance-compression to stance-decompression phase, a change in the rest position of the virtual compliance causes energy to be injected into the spring, initiating push-off. However, due to the geometry of the foot in the shoe, when the knee extends to initiate push-off, only the toe is in contact with the ground and not all of the push-off goes towards lifting the robot off the ground. This is even more pronounced when the spring is close to the rest position. To address this, the switching surface from stance-compression to stance-decompression is modified to

ensure that the transition occurs when the spring is sufficiently compressed,

$$\mathcal{S}_{sc \rightarrow sd}^{exp} := \mathcal{S}_{sc \rightarrow sd} \cup \{x_s \in TQ_s \mid \theta_s > \theta_{50}, q_{Bsp_{st}} < 20^\circ\}, \quad (7.3)$$

where, θ_{50} represents the value of θ_s at 50% into the stance phase.

Modifications to the controller in the stance-decompression phase: During the stance-decompression phase, the designed virtual constraint of the torso specifies the torso to pitch backward with a non-zero velocity. In experiments, due to tracking errors, the torso is sometimes driven forward to reduce tracking errors. This causes the torso to have a forward velocity at lift-off, which causes the torso to pitch forward further in the flight phase, and results in significantly large torso error on impact. To prevent this, the experimental controller is modified such that in the stance-decompression phase, if the torso velocity is below a certain value, the torque is set to a constant value, such that the torso is pushed backward. This essentially means, under certain conditions, the controller for the torso is switched to push the torso backward instead of trying to enforce a virtual constraint.

Modifications to the switching surface $\mathcal{S}_{sd \rightarrow f}$: The following problems are considered. (i) Typically, towards the end of the stance-decompression phase, the ground reaction forces at the stance toe are small as we get closer to a lift-off. Since there is a virtual constraint on the torso, even moderate torques to regulate the torso position towards the end of the stance-decompression phase could cause the foot to slip. We need some way to ensure that the transition occurs prior to the foot slipping. One good way would be to use the spring as an indicator of the ground reaction force. (ii) Further, during the push-off, the stance knee is extending rapidly and the controller needs to prevent the knee from hyper-extending. (iii) If the stance knee is far from hyper-extending, we still cannot have a long duration push-off, since this would result in the hip velocity at take-off being very high, leading to large flight phases during which the torso could pitch forward significantly. These problems are addressed by forcing a transition into the flight phase by the following modification on the stance-decompression to flight switching surface

$$\mathcal{S}_{sd \rightarrow f}^{exp} := \mathcal{S}_{sd \rightarrow f} \cap \{x_s \in TQ_s \mid q_{Bsp_{st}} < 15^\circ, q_{LS_{st}} < 2^\circ\} \cap \{x_s \in TQ_s \mid \dot{p}_{hip}^y > \dot{p}_{hip}^{y,s-*}\}. \quad (7.4)$$

Modifications to the hybrid-invariance controller $\Gamma_f^{\alpha c}$: On initiation of the flight phase, the event-based controller $\Gamma_f^{\alpha c}$ ensures hybrid invariance of the flight phase zero dynamics manifold. This is done through correction polynomials h_c^f as in (6.27). The correction polynomials are employed to smoothly join the original virtual constraint half-way into the flight phase. However, in experiments, certain steps have large errors on initiation of the flight phase. In particular, large errors in the leg angle variables causes the modified leg angle virtual constraints to have an initial reversal in the direction of motion, leading to oscillations, which in turn severely stretches the leg angle cables resulting either in impacts with large errors or the leg angle rapidly shooting towards the hard-stops. To handle large leg angle errors on initiation of the flight phase, the point at which the correction polynomials smoothly join the original virtual constraint is modified adaptively based on the sign and magnitude of the initial error. Errors in one direction are smoothly joined to the original virtual constraint at either 50%, 75%, or 95% into the flight phase based on the magnitude of the error in relation to the total travel, while errors in the opposite direction are always smoothly joined to the original virtual constraint at 50% into the flight phase. This adaptive correction polynomial is instrumental in successfully handling large errors on initiation of the flight phase.

Modifications to the virtual constraints: A final (trivial) modification to the the swing leg shape virtual constraint is performed by commanding the swing leg shape to fold by an additional offset to prevent the toes of the shoes from scuffing the ground during walking and running.

With the above changes to the hardware, and the running controller, the experiment is carried out as follows.

First walking is initiated on this modified system with the same walking controller² developed in Chapter IV. With a torso offset to lean the torso forward, the walking controller induced stable walking at an average speed of 1.26 m/s.

²The impact model for impacts with feet is significantly different from that of a point foot. Double support phases can no longer be approximated to be instantaneous. Despite these significant changes, walking was obtained with the same controller with a clear heel-strike, flat-footed phase, and a toe-off. Thus, with HZD based control design for walking with point feet, walking with non trivial feet was obtained easily. Going the other way: Designing controllers for walking with feet and applying them to systems with point feet may be fairly hard.

To transition from walking to running, a transition controller based on [133] is developed which modifies the virtual constraints of the fixed point for walking to bring the virtual constraint at the end of the walking gait closer to the corresponding virtual constraint at the beginning of the running gait. Instead of a one-step transition from walking to running as done in [85], for MABEL, a two-step transition is carried out. Essentially, there are two transition steps, with the first step being a transition-walk-step, and the second step being a transition-run-step. A walk-to-run transition then consists of the following: A transition from the nominal walking gait to the transition-walk-step, followed by a transition from the transition-walk-step to the transition-run-step, and finally a transition from the transition-run-step to the nominal running gait. This two-step transition enables a smoother transition and prevents rapid torso motions, especially on gaits where the ending and beginning values of the torso virtual constraint differ significantly for the walking and running fixed points respectively. Figure 7.1 illustrates the transition from the transition-walk-step to the transition-run-step.

On transition to running, the running controller is executed. The running controller induced stable running at an average speed of 1.07 m/s. 100 running steps were obtained and the experiment terminated when a legacy feature in the controller transitioned from running to walking at the end of 100 consecutive running steps. Figure 7.2 illustrates snapshots of a typical step of running. Average stance and flight times of 360 ms and 151 ms are obtained respectively, corresponding to a flight phase that is 30% of the gait. An estimated ground clearance of around 2 inches (5 cm) is obtained.

Figure 7.3 illustrates the time evolution of the torso, the B_{spring} pulley, the leg angle and the leg shape coordinates. The location of the squares indicate the instant when the controller switches from the stance phase to the flight phase. The circles on the torso plot indicate the locations at which the controller transitions from stance-compression to stance-decompression subphase. Figure 7.4 illustrates the tracking for the three virtual constraints in the stance phase of running. Recall that in the stance phase, virtual constraints are imposed on the torso, the swing leg angle, and the swing motor leg shape. The stance motor leg shape actuator is not used to impose a virtual constraint, but rather is employed to perform active force control by implementing a virtual compliant element. Figure 7.4



Figure 7.1: Transitioning from walking to running for MABEL with feet. The first step of the transition and the transition from the transition-walk-step to a transition-run-step is shown. Snapshots are at an interval of 100 ms.



Figure 7.2: Typical running step for MABEL with feet. Snapshots are at an interval of 100 ms.

illustrates the tracking for the four virtual constraints in the flight phase of running. Recall that in the flight phase, virtual constraints are imposed on the stance (old stance) leg angle and motor leg shape, the swing absolute leg angle, and the swing motor leg shape. Figure 7.6 illustrates the corresponding motor torques. Figure 7.7 illustrates the cable stretch in the leg shape direction. A peak of approximately 8° in the leg shape coordinates is observed. This corresponds to the knee bending by 16° due to cable stretch. The active force control is able to account for this significant cable stretch. Finally, Figure 7.8 illustrates the β and γ -parameters for 70 steps of running.

7.1.2 Running with Point Feet

The insight garnered from obtaining successful steady-state running with passive feet is used to implement a running controller when the robot is equipped with point feet.

The following additional modifications to the running controller presented in Chapter VI are performed. As suggested in the previous section, for speed regulation, the γ -parameter corresponding to the virtual compliance stiffness is modified as in (7.1). The saturation for β -parameter corresponding to the touch down angle is modified as in (7.2). The saturation for the γ -parameter that modifies the location of the stance-compression to stance-decompression phase is also modified as a function of the speed and is given as,

$$\gamma_{\delta_{sc \rightarrow sd}}^{sat} = \begin{cases} 0.2, & 0 \leq sp(x_s^+, x_s^{+prev}) < 2 \\ 0.25, & 2 \leq sp(x_s^+, x_s^{+prev}) < 2.5 \\ 0.35, & 2.5 \leq sp(x_s^+, x_s^{+prev}) \end{cases} \quad (7.5)$$

At high speeds, the time spent in the stance-decompression phase decreases which results in less energy being injected and smaller push-offs. With the above modification, a well defined flight phase is maintained even when running very fast.

The stance-compression to stance-decompression transition is not modified. During the stance-decompression phase, the torso is pushed back in a similar manner as in the running with feet experiment. To prevent the stance-decompression phase from causing a lift-off with a high velocity, the stance-decompression to flight phase switching surface is modified

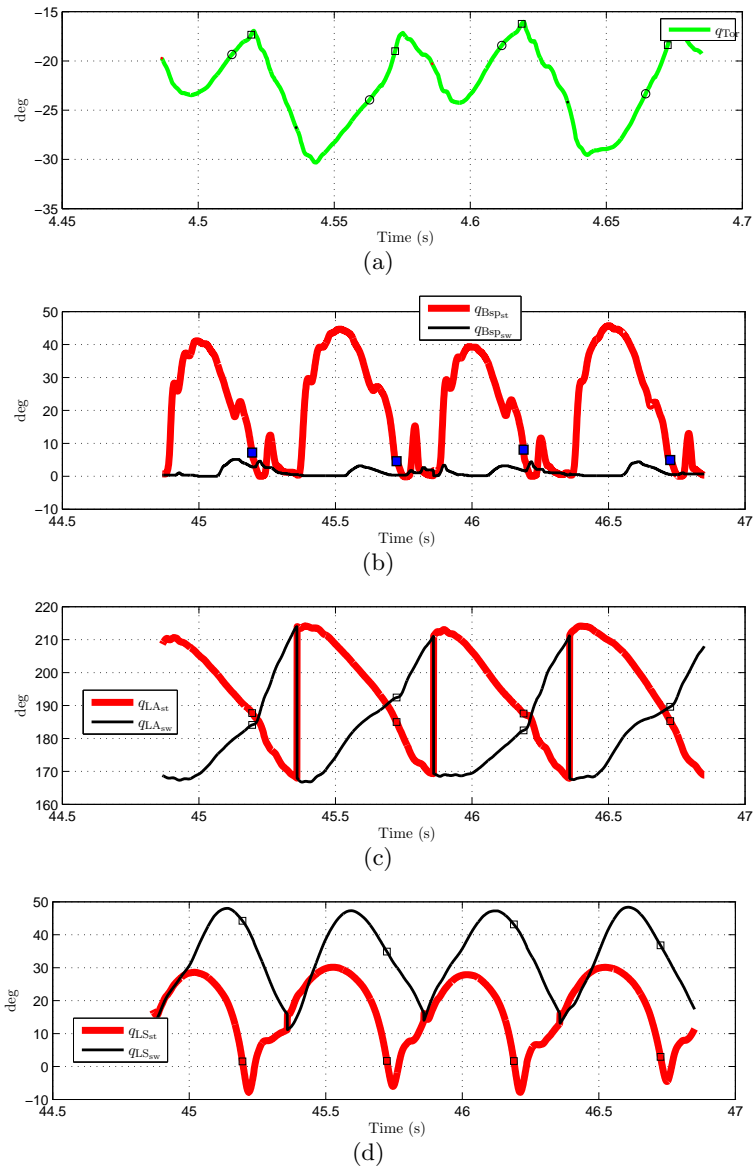


Figure 7.3: Evolution of the following coordinates for four steps of the running experiment with feet: (a) Torso, (b) B_{spring} , (c) Leg angle, and (d) Leg shape. The squares indicate the locations at which the controller transitions from the stance phase to the flight phase. The circles on the torso plot indicate the locations at which the controller transitions from the stance-compression subphase to the stance-decompression subphase. The black dots on the torso plot indicate impact on the left (inner) leg, while red dots indicate impact on the right (outer) leg.

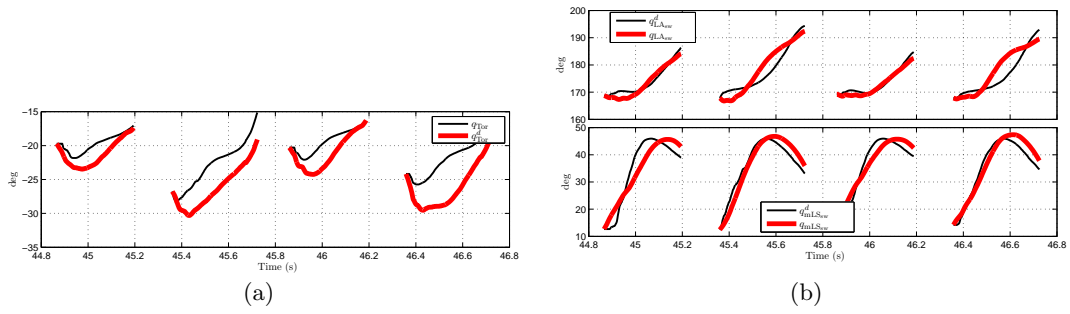


Figure 7.4: Tracking for the virtual constraints in the stance phase of running. Recall that the stance phase has three virtual constraints imposed on the torso, and the swing leg angle and the swing motor leg shape. The swing motor leg shape plot is graphed by scaling into the leg shape coordinates.

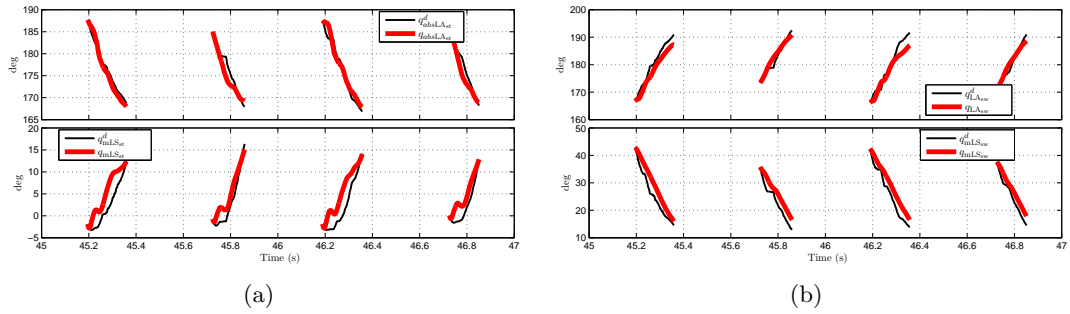


Figure 7.5: Tracking for the virtual constraints in the flight phase of running. Recall that the flight phase has four virtual constraints imposed on the stance (old stance) leg angle and motor leg shape, and the swing absolute leg angle and the swing motor leg shape.

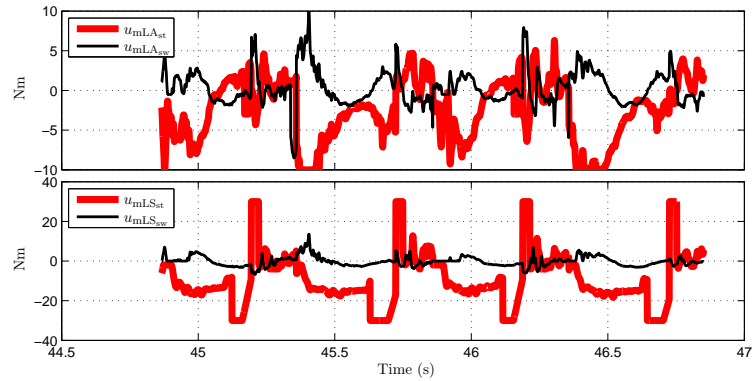


Figure 7.6: Motor torques for the running experiment with feet.

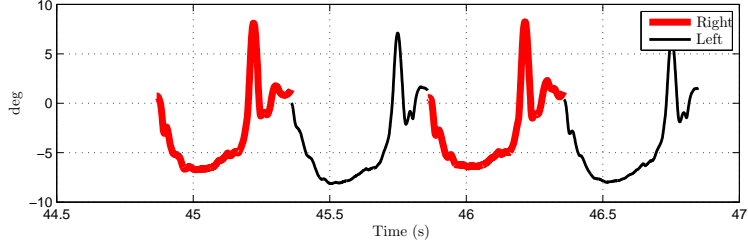


Figure 7.7: Cable stretch in the leg shape direction for the stance leg.

as follows

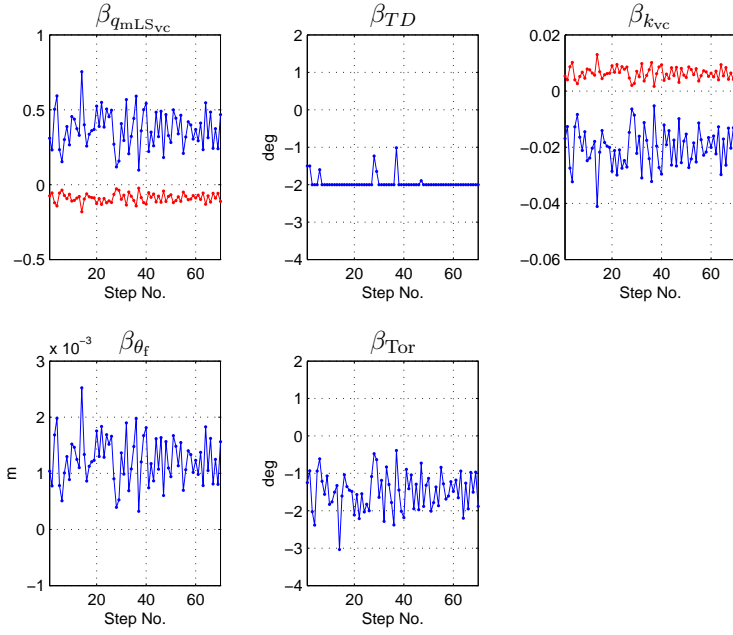
$$\mathcal{S}_{sd \rightarrow f}^{exp} := \mathcal{S}_{sd \rightarrow f} \cap \{x_s \in TQ_s \mid \dot{p}_{\text{hip}}^y > \dot{p}_{\text{hip}}^{y,s-*}\}. \quad (7.6)$$

During the flight phase, the adaptive correction polynomials as used for the running with feet experiment are deployed.

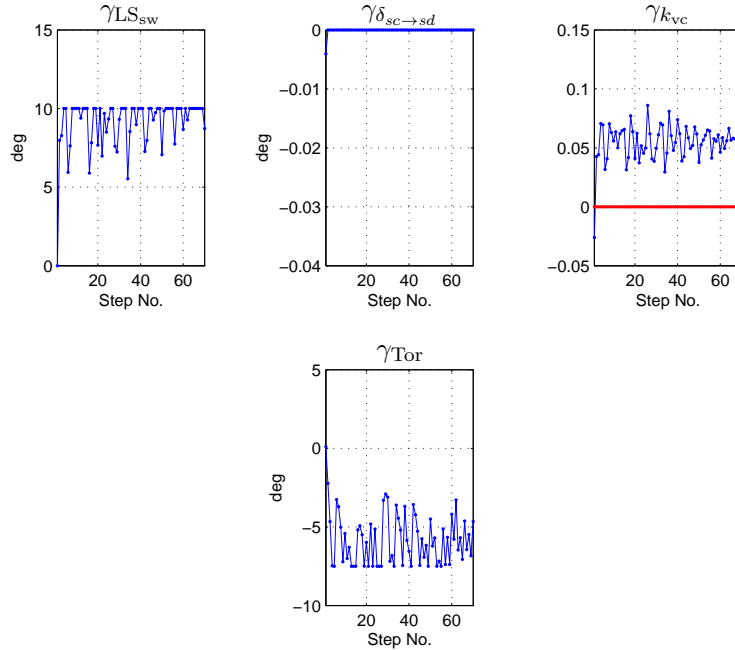
With these changes to the controller developed in Chapter VI, the experiment with point feet is carried out in a similar procedure as in the experiment with non-trivial feet. The running controller induced stable running at an average speed of 1.95 m/s, and a peak speed of 3.06 m/s. 113 running steps were obtained and the experiment terminated when the power to the robot was cut off. At 2 m/s, the average stance and flight times of 233 ms and 126 ms are obtained respectively, corresponding to a flight phase that is 35% of the gait. At 3 m/s, the average stance and flight times of 195 ms and 123 ms are obtained respectively, corresponding to a flight phase that is 39% of the gait. An estimated ground clearance of 3 – 4 inches (7.5 – 10 cm) is obtained.

Figure 7.9 depicts snapshots at 100 ms intervals of a transition from walking to running. Figure 7.10 depicts snapshots at 100 ms intervals of a typical running step.

Figure 7.11 illustrates the speeds for each walking and running step. The robot transitions into running on step 21. The outer-loop event based controller parameters are depicted in Figure 7.12. There is considerable variation in speed. When the speed exceeds 2.5 m/s, large changes in the touch down angle, β_{TD} , and the γ -parameter that affects the transition from stance-compression to stance-decompression, $\gamma_{\delta_{sc \rightarrow sd}}$ causes the speed to dramatically drop to under 1 m/s. All this is autonomously handled by the controller with no manual intervention. The ability of the controller to recover from slow speeds below 1 m/s, and high speeds above 2.5 m/s illustrates a large domain of attraction of the designed controller.



(a)



(b)

Figure 7.8: Parameter plots for the outer-loop controllers employed in running with feet. (a) Parameters for the Γ^β exponentially stabilizing outer-loop controller, (b) Parameters for the Γ^γ domain of attraction enlarging outer-loop controller.

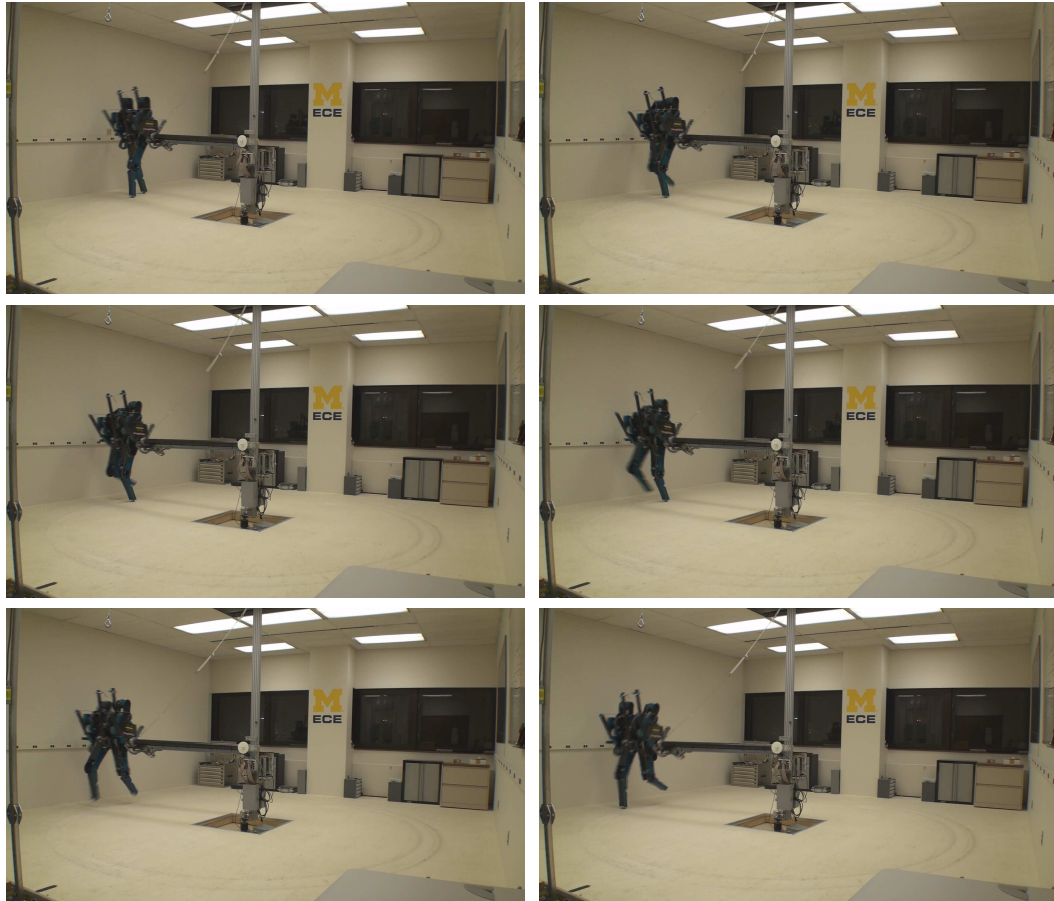


Figure 7.9: Transitioning from walking to running for MABEL with point feet. The second step of the transition which transitions from the transition-walk-step to a transition-run-step is shown. Snapshots are at an interval of 100 ms.

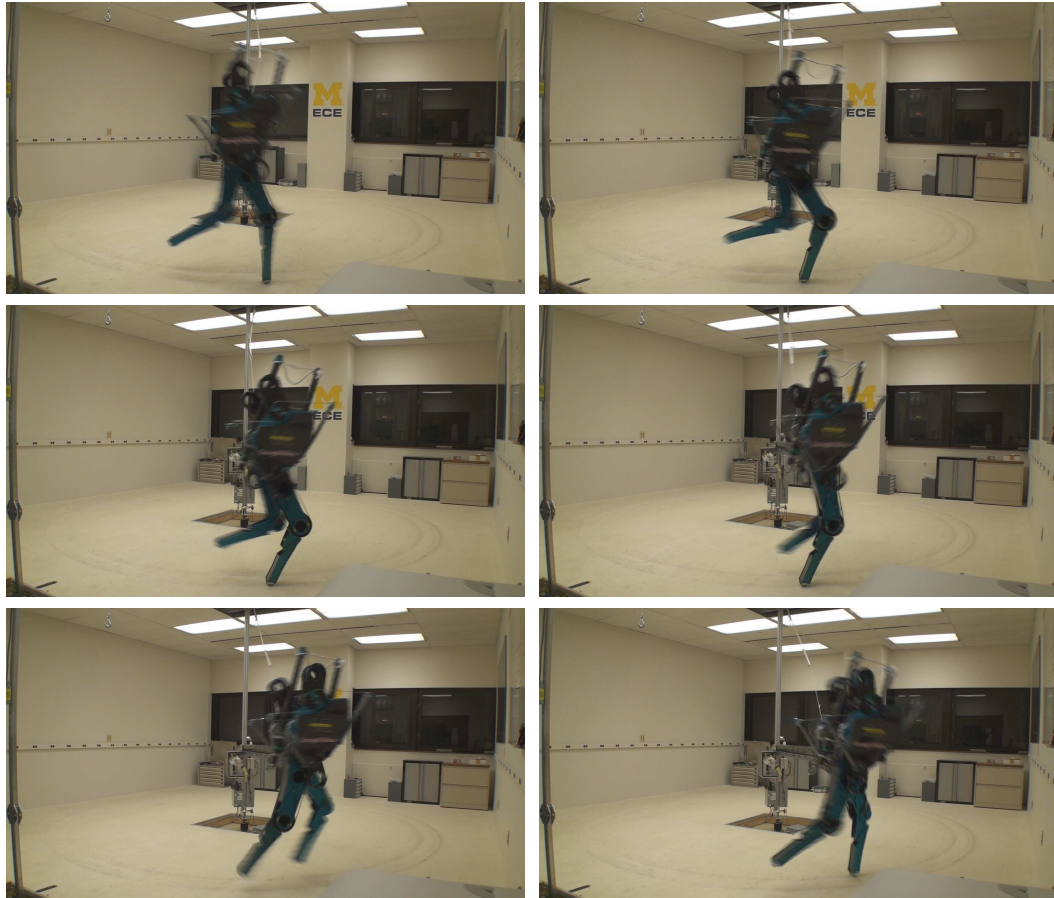


Figure 7.10: Typical running step for MABEL with point feet. Snapshots are at an interval of 100 ms.

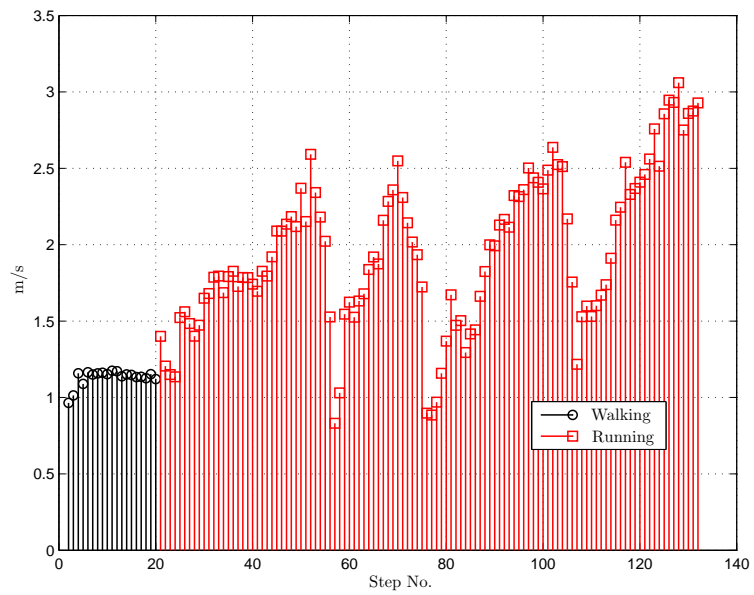
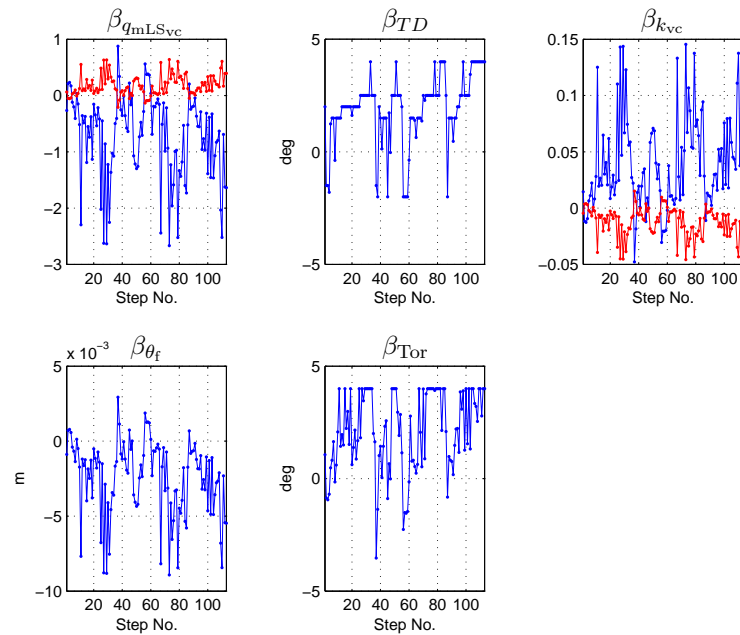
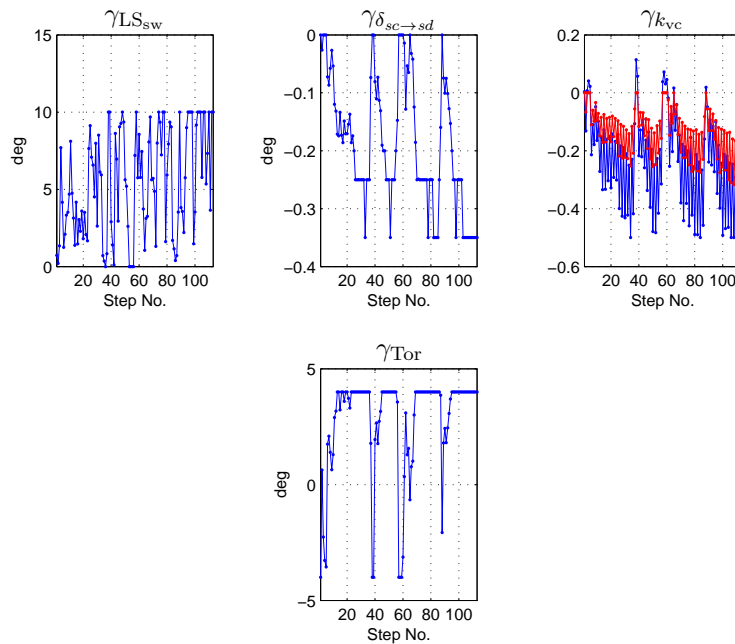


Figure 7.11: Speed at each step for the running experiment with point feet. The black lines with circles represent the walking steps, while the red lines with the squares represent the running steps.



(a)



(b)

Figure 7.12: Parameter plots for the outer-loop controllers employed in running. (a) Parameters for the Γ^β exponentially stabilizing outer-loop controller, (b) Parameters for the Γ^γ domain of attraction enlarging outer-loop controller.

CHAPTER VIII

Concluding Remarks

8.1 Summary of New Contributions

MABEL contains springs in its drivetrain for the purposes of enhancing energy efficiency and agility of dynamic locomotion. This thesis has presented a novel analytical design method to realize the potential of the springs. An extensive set of experiments have been performed to illustrate and confirm important aspects of the feedback design.

A HZD-based controller was designed to achieve asymptotically stable walking while recruiting the compliance in the robot's drivetrain to perform most of the negative work required to decelerate the downward motion of the robot's center of mass after impact, instead of the actuators. This not only improved the energy efficiency of walking, but also made the gait more natural looking. Stability analysis of the walking gait was performed using the method of Poincaré.

The analytically derived control law was experimentally validated on MABEL. The controller was demonstrated to be robust to external disturbances as well as to significant differences between the design model and the actual robot. In particular, the cables used to realize the differentials in the robot's drivetrain exhibited considerable stretch in the experiments, none of which was considered in the design model. Due to the observations made as part of these experiments, a more accurate model incorporating cable stretch has been presented in [50].

A walking gait was designed to optimize for the energetic cost of mechanical transport c_{mt} and then experimentally evaluated on MABEL. Even though MABEL has no feet, the

experimentally realized c_{mt} is 12 times better than that of ASIMO, approximately three times as good as RABBIT, and twice as good as a hand-designed virtual-constraint-based controller that we had previously implemented on MABEL. This puts MABEL's energy efficiency within a factor of two of T.U. Delft's Denise, and a factor of three of the Cornell Biped, which are specifically designed mechanically for efficient walking. This demonstrates the interplay of mechanical design and control design in achieving higher efficiency.

For the first time, a real-time implementation of a complete hybrid zero dynamics based controller has been demonstrated in experiments. The tracking accuracy attained is far better than that of simple PD controllers used in prior experiments on RABBIT and MABEL. This removed the restriction of hybrid zero dynamics to theory or simulation, and establishes hybrid zero dynamics based controllers in the experimental domain.

A controller was implemented on MABEL that realized a sustained walking speed of 1.5 m/s (3.4 mph). This made MABEL "the fastest robotic bipedal walker *of any size*" as of October 31, 2009, and the record was held until April 22, 2010.

A running control design is presented that creates a compliant and actuated hybrid zero dynamics during the stance phase of running, enabling active force control. With force control, a virtual compliant element is created to enable dynamically varying the effective leg compliance during stance. An outer-loop event-based controller has been designed to exponentially stabilize the periodic running gait. An additional outer-loop event-based controller has been designed to enlarge the domain of attraction of the periodic running gait.

The running controller has been experimentally deployed and stable running has been successfully demonstrated on MABEL, both with passive feet and with point feet. The achieved running is dynamic and life-like with flight phases of significant duration and high ground clearance. For running with point feet, a peak speed of 3.06 m/s (6.8 mph) was obtained, making MABEL the fastest kneed bipedal runner as of July 30, 2011.

8.2 Perspectives on Future Work

Moving Heuristics to Analysis: The running controller developed has multiple loops, Γ^α , Γ^{α_c} , Γ^β , and Γ^γ . The inner-loops, Γ^α , Γ^{α_c} , and Γ^β , are auto designed using rigorous analytical techniques. Γ^α , and Γ^β are partly driven by the morphology of the bipedal robot since the choice of the virtual constraints and the β -parameters is partly dependent on the morphology. The outer-loop event-based controller, Γ^γ , is however essentially based on heuristics. The question that remains unanswered is, can this outer-loop controller be analytically designed?

Firstly, even in the absence of Γ^γ , the designed controller creates an exponentially stable periodic orbit. The outer-loop event-based controller Γ^γ is added to increase the domain of attraction of the designed running controller for experimental deployment, and to provide the capability to address issues that arise due to the discrepancy between the model and the physical system, such as the issue of speed regulation. If the design model matched more closely with the physical system, speed regulation would not be an issue, as it would have been taken care by the exponentially stabilizing event-based controller Γ^β . Nevertheless, analytically estimating the domain of attraction for a periodic solution for a complex system such as MABEL, and then designing controllers to increase this domain, is an extremely hard problem.

One way to analytically design the final layer of the controller would be to come up with the reduced order system, or some mechanical analog, representing the zero dynamics. Once such a system is available, then analytical methods can be used to design a controller for this system. However, coming up with the reduced order system is non-trivial. One way to analytically design a controller to perform speed regulation would be to do carry out energy regulation [92]. However, this requires estimating the kinetic and potential energies, typically very noisy quantities in real experiments. Thus, the problem of analytically designing a practical Γ^γ domain of attraction enlarging controller is still an open problem and needs to be investigated further.

Running over Rough Terrain: A natural extension of the work presented in this thesis would be to demonstrate running over rough terrain. Since the framework of active

force control with the compliant hybrid zero dynamics has been setup to be able to vary the effective leg stiffness, running experiments over rough terrain could be performed. As suggested in the biomechanics literature, by varying the leg stiffness, robust running over rough terrain may be obtained.

Finding Periodic Orbits: One weakness of the hybrid zero dynamics framework is the computation of periodic solutions. On MABEL, this was fairly involved for walking motions, and really hard for obtaining periodic running motions. For future 3D robots with compliance, which are of higher-dimension compared to MABEL, carrying out an optimization process may be extremely computationally expensive. Efficient methods and optimization techniques to find periodic orbits need to be formulated.

APPENDICES

APPENDIX A

Bézier Polynomials for Subphases

This section develops a framework for virtual constraints with subphases and provides details on how the Bézier polynomials for the stance subphases are obtained.

Framework for Virtual Constraints with Subphases

Let \mathcal{P} be an index set representing the subphases of the virtual constraints, and let $p \in \mathcal{P}$ denote a particular subphase. Similarly, let \mathcal{V} be an index set representing the virtual constraints, and let $v \in \mathcal{V}$ denote a particular virtual constraint. In each subphase, the virtual constraint will be an M^{th} order Bézier polynomial parametrized by θ_s , where, θ_v^{p+} , θ_v^{p-} are the starting and ending values of θ_s respectively within phase p of virtual constraint v . It is convenient to normalize each θ_s to $[0, 1]$ by defining

$$s_v^p := \frac{\theta_s - \theta_v^{p+}}{\theta_v^{p-} - \theta_v^{p+}}, \quad (\text{A.1})$$

where θ_s goes from θ_v^{p+} to θ_v^{p-} during phase $p \in \mathcal{P}$. Then, if α_v^p is the vector of Bézier coefficients, then the desired evolution of the virtual constraint v for phase p can be expressed as

$$h_v^{d,p}(\theta_s) = \sum_{k=0}^M \alpha_v^p(k) \frac{M!}{k!(M-k)!} s_v^{pk} (1-s_v^p)^{M-k}. \quad (\text{A.2})$$

To ensure \mathcal{C}^k continuity between successive subphases $p_1, p_2 \in \mathcal{P}$, a standard property of Bézier curves specifies how the last $k+1$ parameters of $\alpha_v^{p_1}$ and the first $k+1$ parameters of

$\alpha_v^{p_2}$ must be related [130, p. 139]. Next, it can be convenient to treat successive subphases p_1, p_2 , defined over domains $[\theta_v^{p_1+}, \theta_v^{p_1-}]$, $[\theta_v^{p_2+}, \theta_v^{p_2-}]$ respectively, as a single combined phase $p_1 p_2$ with domain $[\theta_v^{p_1+}, \theta_v^{p_2-}]$. This is particularly useful for virtual constraints that do not require the resolution of multiple subphases and enables parameterizing with a single Bézier polynomial over both subphases. To do this, we impose $\theta_v^{p_1-} = \theta_v^{p_2-}$, $\theta_v^{p_2+} = \theta_v^{p_1+}$, and $\alpha_v^{p_1} = \alpha_v^{p_2} =: \alpha_v^{p_1 p_2}$ in our general framework for virtual constraints with subphases. Thus we obtain $s_v^{p_1 p_2}|_{p_1} = s_v^{p_1}$ and $s_v^{p_1 p_2}|_{p_2} = s_v^{p_2}$, with $s_v^{p_1 p_2}$ monotonically increasing from zero to one over both phases p_1, p_2 . With this setup, the transition event from subphase p_1 to p_2 no longer has any effect on the combined phase $p_1 p_2$. This phase combination can be extended to more than two phases, and in fact all of the phases of a controlled variable can be combined, resulting in a single virtual constraint over the entire stance phase with no subphases.

Stance Motor Leg-shape Virtual Constraint

In the motor-compression phase, the motor leg-shape position is given by a Bézier polynomial parametrized by $s_{\text{mLS}_{\text{st}}}^{mc}$, with coefficients $\alpha_{\text{mLS}_{\text{st}}}^{mc}$ and $\theta_{\text{mLS}_{\text{st}}}^{mc+} = \theta^{s+}$, $\theta_{\text{mLS}_{\text{st}}}^{mc-} = \theta^{mc-}$. The boundary conditions ($q_{\text{mLS}_{\text{st}}}^{mc+} = q_{\text{mLS}_{\text{st}}}^{s+}$, $\dot{q}_{\text{mLS}_{\text{st}}}^{mc+} = \dot{q}_{\text{mLS}_{\text{st}}}^{s+}$), and ($q_{\text{mLS}_{\text{st}}}^{mc-} = q_{\text{mLS}_{\text{st}}}^{sc}$, $\dot{q}_{\text{mLS}_{\text{st}}}^{mc-} = 0$) specify the starting and the ending two coefficients, and the middle coefficients are free to be chosen as part of the control design, and are chosen to smoothly transition between $q_{\text{mLS}_{\text{st}}}^{s+}$ and $q_{\text{mLS}_{\text{st}}}^{sc}$.

In the stance-compression phase, the motor leg-shape position is given by a Bézier polynomial parametrized by $s_{\text{mLS}_{\text{st}}}^{sc}$, with coefficients $\alpha_{\text{mLS}_{\text{st}}}^{sc}$ all equal to $q_{\text{mLS}_{\text{st}}}^{sc}$, and $\theta_{\text{mLS}_{\text{st}}}^{sc+} = \theta^{s+}$, $\theta_{\text{mLS}_{\text{st}}}^{sc-} = \theta^{sc-}$.

In the stance-injection phase, the motor leg-shape position is given by a Bézier polynomial parametrized by $s_{\text{mLS}_{\text{st}}}^{si}$, with coefficients $\alpha_{\text{mLS}_{\text{st}}}^{si}$ and $\theta_{\text{mLS}_{\text{st}}}^{si+} = \theta^{si+}$, $\theta_{\text{mLS}_{\text{st}}}^{si-} = \theta^{si-}$. The boundary conditions ($q_{\text{mLS}_{\text{st}}}^{si+} = q_{\text{mLS}_{\text{st}}}^{sc}$, $\dot{q}_{\text{mLS}_{\text{st}}}^{si+} = 0$), ($q_{\text{mLS}_{\text{st}}}^{si-} = q_{\text{mLS}_{\text{st}}}^{sc-}$, $\dot{q}_{\text{mLS}_{\text{st}}}^{si-} = 0$) specify the starting and ending two parameters of $\alpha_{\text{mLS}_{\text{st}}}^{si}$ with the rest being free parameters to be chosen as part of control design.

In the stance-decompression phase, the motor leg-shape position is given by a Bézier

polynomial parametrized by $s_{\text{mLS}_{\text{st}}}^{sd}$, with coefficients $\alpha_{\text{mLS}_{\text{st}}}^{sd}$ all equal to $q_{\text{mLS}_{\text{st}}}^{sc-}$, and $\theta_{\text{mLS}_{\text{st}}}^{sd+} = \theta^{sd+}$, $\theta_{\text{mLS}_{\text{st}}}^{sd-} = \theta^{s-}$.

Thus, parameters of only $\alpha_{\text{mLS}_{\text{st}}}^{si}$ are available to be chosen as part of control design. For notation purposes, we define $\alpha_{\text{mLS}_{\text{st}}} := \alpha_{\text{mLS}_{\text{st}}}^{si}$.

Torso Virtual Constraint

The motor-compression, stance-compression, and stance-injection phases, are combined into a single phase by setting $\theta_{\text{Tor}}^{mc+} = \theta_{\text{Tor}}^{sc+} = \theta_{\text{Tor}}^{si+} = \theta^{s+}$, $\theta_{\text{Tor}}^{mc-} = \theta_{\text{Tor}}^{sc-} = \theta_{\text{Tor}}^{si-} = \theta^{s-}$, and $\alpha_{\text{Tor}}^{mc} = \alpha_{\text{Tor}}^{sc} = \alpha_{\text{Tor}}^{si} =: \alpha_{\text{Tor}}$. Thus the torso evolution in this combined phase is given by a Bézier polynomial parametrized by s_{Tor} , with coefficients α_{Tor} .

The desired torso evolution in the stance-decompression phase is given by a Bézier polynomial parametrized by s_{Tor}^{sd} , with coefficients α_{Tor}^{sd} all equal to q_{Tor}^{s-} , and $\theta_{\text{Tor}}^{sd+} = \theta^{sd+}$, $\theta_{\text{Tor}}^{sd-} = \theta^{s-}$.

Swing Leg Virtual Constraints

For the swing virtual constraints, all subphases are combined into one by setting $\theta_{\text{LA}_{\text{sw}}}^{mc+} = \theta_{\text{LA}_{\text{sw}}}^{sc+} = \theta_{\text{LA}_{\text{sw}}}^{si+} = \theta_{\text{LA}_{\text{sw}}}^{sd+} = \theta^{s+}$, $\theta_{\text{LA}_{\text{sw}}}^{mc-} = \theta_{\text{LA}_{\text{sw}}}^{sc-} = \theta_{\text{LA}_{\text{sw}}}^{si-} = \theta_{\text{LA}_{\text{sw}}}^{sd-} = \theta^{s-}$, and $\alpha_{\text{LA}_{\text{sw}}}^{mc} = \alpha_{\text{LA}_{\text{sw}}}^{sc} = \alpha_{\text{LA}_{\text{sw}}}^{si} = \alpha_{\text{LA}_{\text{sw}}}^{sd} =: \alpha_{\text{LA}_{\text{sw}}}$. Thus the evolution of the swing leg angle is given by a Bézier polynomial parametrized by $s_{\text{LA}_{\text{sw}}}$, with coefficients $\alpha_{\text{LA}_{\text{sw}}}$. In a completely similar manner, the swing motor leg shape is parameterized by $\alpha_{\text{mLS}_{\text{sw}}}$.

The Bézier coefficients that are not specified above are free parameters in the virtual constraints, and are specified by control design. These parameters can be put in a vector as,

$$\alpha_{\text{s}} = \begin{bmatrix} \alpha_{\text{mLS}_{\text{st}}} \\ \alpha_{\text{LA}_{\text{sw}}} \\ \alpha_{\text{mLS}_{\text{sw}}} \\ \alpha_{\text{Tor}} \end{bmatrix}. \quad (\text{A.3})$$

APPENDIX B

Optimization Details for Walking

Equality and inequality constraints are used during the optimization process to ensure that the closed-loop system yields a desired behavior. These constraints could be limits on peak actuator torques, joint space constraints, unilateral ground contact forces, speed of walking, ground clearance, etc. Further, the general form of the virtual constraints chosen in Section 4.1 is assumed to be satisfied on the periodic orbit. This enables integrating the stance zero dynamics over the reduced-order closed-loop system dynamics (established in Section 4.3) thereby reducing the computation time significantly. The details of the optimization algorithm are described below.

Algorithm

1. Select θ_s^{-*} , $q_{\text{Bsp}_{\text{st}}}^{-*}$, $q_{\text{LA}_{\text{sw}}}^{-*}$, $q_{\text{mLS}_{\text{sw}}}^{-*}$, q_{Tor}^{-*} . Determine $q_{\text{LA}_{\text{st}}}^{-*}$ using (4.4). Determine $q_{\text{mLS}_{\text{st}}}^{-*}$ by a Newton-Rhapson search to satisfy $p_{\text{toe}_{\text{sw}}}^{v^{-*}} = 0$ as this ensures that the impact condition, $q^{-*} \in S_{\text{s} \rightarrow \text{s}}$, is met.
2. Select $\dot{\theta}_s^{-*}$, $\dot{q}_{\text{Bsp}_{\text{st}}}^{-*}$, $\dot{q}_{\text{LA}_{\text{sw}}}^{-*}$, $\dot{q}_{\text{mLS}_{\text{sw}}}^{-*}$. Choose $\dot{q}_{\text{mLS}_{\text{st}}}^{-*} = 0$, $\dot{q}_{\text{Tor}}^{-*} = 0$ to satisfy the virtual constraints described in section 4.1. Determine $\dot{q}_{\text{LA}_{\text{st}}}^{-*}$ using (4.4).
3. Using the stance-to-stance transition function, $\Delta_{\text{s} \rightarrow \text{s}}$, obtain $x_s^{+*} = (q_s^{+*}; \dot{q}_s^{+*})$, the state corresponding to the beginning of the subsequent stance phase.
4. Calculate θ_s^{+*} , $\dot{\theta}_s^{+*}$ using (4.4). Set $\theta_{\text{mc}}^{-*} = \theta_s^{+*} + 0.05(\theta_s^{-*} - \theta_s^{+*})$, and $\theta_{\text{si}}^{-*} = \theta_s^{+*} + 0.7(\theta_s^{-*} - \theta_s^{+*})$ corresponding to 5% and 70% of the stance phase respectively, and

set $q_{\text{mLS}_{\text{st}}}^{\text{sc}*} = q_{\text{mLS}_{\text{st}}}^{\text{s+}*} + \text{sgn}(\dot{q}_{\text{mLS}_{\text{st}}}^{\text{s+}*}) \max(20^\circ, |q_{\text{mLS}_{\text{st}}}^{\text{s+}*}|)$. This facilitates the use of the first 5% of the gait to drive non-zero post-impact motor leg shape velocity to zero, and the last 30% of the gait to hold the torso constant in preparation for impact.

5. Select $\alpha_{\text{s},2}^*, \dots, \alpha_{\text{s},M-2}^*$. Calculate $\alpha_{\text{s},0}^*, \alpha_{\text{s},1}^*$ to satisfy the post-impact conditions, and calculate $\alpha_{\text{s},M-1}^*, \alpha_{\text{s},M}^*$ to satisfy the pre-impact conditions. Set $\alpha_{\text{mLS}_{\text{st},2}}^{\text{mc}*}, \dots, \alpha_{\text{mLS}_{\text{st},M-2}}^{\text{mc}*}$ to get a smooth transition between $q_{\text{mLS}_{\text{st}}}^{\text{s+}*}$ and $q_{\text{mLS}_{\text{st}}}^{\text{sc}*}$.
6. Integrate the stance dynamics for the motor-correction phase, Σ_{mc} , and the stance-compression phase, Σ_{sc} , until the spring undergoes maximum compression and reaches a decompressed value of five degrees. Set this value of θ_{s} as $\theta_{\text{s}}^{\text{sc}-}$. Integrate the stance dynamics through the stance-injection, Σ_{si} , and stance-decompression, Σ_{sd} , phases to obtain x_{s}^- .
7. Evaluate the cost function J , equality constraints EQ , inequality constraints $INEQ$.
8. Iterate the above steps until J is minimized and the equality and inequality constraints are satisfied.

Equality constraints, EQ

- Error associated with finding a fixed point $\|x_{\text{s}}^- - x_{\text{s}}^{-*}\| = 0$.
- Toe position of the swing leg at the end of the step $p_{\text{toe}_{\text{sw}}}^v = 0$.

Inequality constraints, $INEQ$

- Magnitude of the minimum normal force at the stance leg to be positive, $\min(F_{\text{st}}^N) > 0$.
- Maximum of magnitude of coefficient of friction less than one, $|\max(F_{\text{st}}^T/F_{\text{st}}^N)| < 0.6$.
- Walking speed greater than 0.7 m/s.
- Swing leg toe profile to be above the ground throughout the stance phase.
- Swing leg angle not to exceed 220° .

Table B.1: The list of independent parameters to be determined by optimization. The choice of these parameters is non-unique, and depends on the algorithm and constraints employed in optimization.

Optimization Parameters

$$\begin{aligned}
 \theta_s^{-*}, q_{\text{Bspst}}^{-*}, q_{\text{LA}_{\text{sw}}}^{-*}, q_{\text{mLS}_{\text{sw}}}^{-*}, q_{\text{Tor}}^{-*} &\in \mathbb{R} \\
 \dot{\theta}_s^{-*}, \dot{q}_{\text{Bspst}}^{-*}, \dot{q}_{\text{LA}_{\text{sw}}}^{-*}, \dot{q}_{\text{mLS}_{\text{sw}}}^{-*} &\in \mathbb{R} \\
 \alpha_2^{s*}, \dots, \alpha_{M-2}^{s*} &\in \mathbb{R}^4
 \end{aligned}$$

- Stance leg angle not less than 140° .
- Range of travel of torso less than 5° .

APPENDIX C

Robot Construction and System ID

This chapter briefly discusses the behind the scenes work that went on in constructing the bipedal testbed at The University of Michigan, and summarizes the system identification work that was carried out to identify the eleven degree model of MABEL.

Robot Construction

When the author first began this work, the design of the bipedal system had terminated, and various parts were being fabricated. The electronic system was being prototyped and a design for the various PCB boards was being established. The bipedal lab at The University of Michigan had just a new floor and lots of unopened boxes of various robot parts. Figure C.1 illustrates the status as on August 2007. A frenzied collaborative activity began, between the author and colleagues, leading to the mechanically assembling the robot, testing of motor amplifiers, prototyping of encoder interfaces, fabrication, population, and debugging of electronic PCBs, wiring of the entire bipedal testbed system, development of a real-time data acquisition software, and finally the preliminary testing that ensured the system was working. This activity concluded in March 2008, and Figure C.2 illustrates the completed testbed. This process had to successfully terminate before being able to run any soft of controllers for achieving walking.



(a)



(b)

Figure C.1: The status of the bipedal testbed when the author began work in August 2007 (a) the empty lab, (b) various robot parts to be assembled.

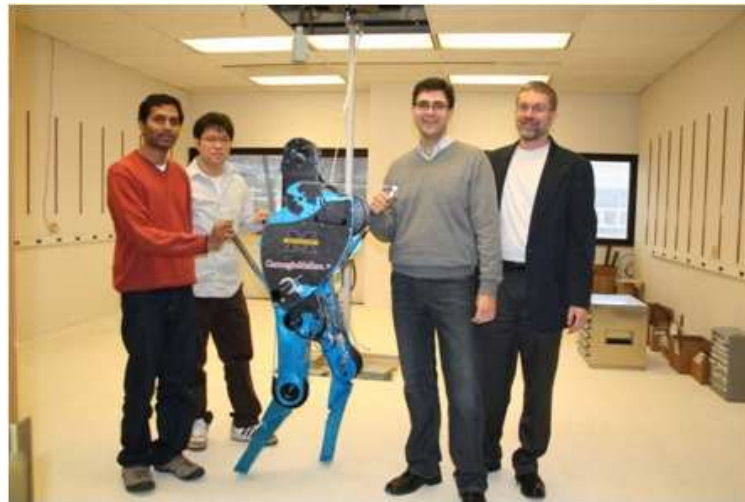


Figure C.2: Mechanical assembly of MABEL, all electronic component and wiring successfully completed by March 2008 and the testbed was ready for trying out controllers. (From Left to Right, The author, Hae-Won Park, MABEL, Prof. Ioannis Poulakakis, and Prof. Jessie W. Grizzle.)

System Identification

The system identification of a 5-link bipedal robot with compliant transmission has been performed, as described in [50], to arrive at a eleven degree of freedom model for MABEL. For each side of the robot, the transmission comprises of three cable differentials that connect two actuators to indirectly actuate the hip and thigh joints in such a way that one actuator controls the angle of the virtual leg consisting of the line connecting the hip to the toe, and the second actuator controls the length of the virtual leg. The differentials also place a compliant element in series with the actuator controlling the virtual leg length.

The robot is equipped with fifteen encoders to measure motor, pulley and joint angles, as well as contact switches to sense foot contact, and joint configuration limits, and two laser sensors to accurately measure hip height and torso angle. Neither force sensors, torque sensors nor accelerometers are available on the robot, and model parameters need to be identified on the basis of these measurements and commanded torques to the four actuators. To get around these limitations, the identification procedure took full advantage of the modular nature of the robot. By selectively disconnecting the cables in the transmission, various elements can be isolated for identification.

The model parameters consist of the actuator parameters (rotor inertia and torque constants), the friction coefficients and inertias for the various joints and pulley, and the spring constants for the compliant elements. An initial *a priori* estimate of the robot's overall dynamic model can be obtained from CAD. So system identifications consists of finding parameters that are not part of the model (such as the actuator parameters, the compliance constants, etc.), and improving the CAD estimates for inertias and masses of the various components making up the transmission and linkage.

The inertia parameters are identified in a modular fashion by starting with just the actuator and commanding a torque signal, and then following it up with experiments that sequentially connect more cables of the transmission to include more pulleys and joints. This phase of the system has all compliant elements disconnected.

Following this, the spring constants for the compliant elements are estimated through a static experiment. As part of various experiments, it was discovered that MABEL has

another source of compliance - the cables. Cables that connect different parts of the transmission stretch significantly during heavy loads. The cable stretch can be measured due to the redundant encoders present on the pulleys and joints, and was identified from a set of hopping experiments.

A complete dynamic model of the robot is constructed using the parameters identified in the above process. This overall model of the robot has been validated through a dynamic hopping experiment that excites all of the dynamics of the model.

Two Models for Design and Verification : Simple vs Complex Model

The model of the robot presented in Section 3.2 makes certain key assumptions: (a) The ground is modeled as a rigid ground, (b) The cables in the transmission of the robot are assumed to not stretch, (c) The robot is assumed to be planar. This model is a *simple model* that is used for all control design in this thesis. Walking controllers designed based on the simple model worked relatively well in experiments. However, all attempts at designing a hopping controller based on the simple model and experimentally validating the control design failed. The floor in the lab has a layer of rubber, and modeling this as a rigid ground is inappropriate, especially for hopping gaits where there are large impacts with the ground. Further in hopping experiments the cables stretch even more than that in walking and can not be ignored. Finally, due to the hip distance between the legs of the robot being 10% of the length of the boom, the robot weighed 10% heavier on the inner leg - almost 7 Kg. Thus, a *complex model* that dropped the above assumptions was needed. In particular the complex model includes: (a) A compliant ground, (b) Stretchy cables, (c) A $2\frac{1}{2}$ D model of the robot. A complete system identification of MABEL, development of the complex model and a hopping experiment validation using the complex model was briefly discussed in the previous section and is presented in detail in [50].

APPENDIX D

Improving Energy Efficiency Further

The energy efficient walking gait obtained by choosing the specific cost of mechanical transport, c_{mt} , as the cost function improves the c_{mt} of the nominal fixed point by 10%. As illustrated in Section 5.1.3 an equivalent improvement was observed in the experiments. However there is a big discrepancy of over 350% between the model predicted value and the experimentally obtained value of c_{mt} . In this section we present changes in the model that bring down this large discrepancy down to 10%. We then investigate the effect of certain controller parameters on energy efficiency to find a way to further improve the energy efficiency of the efficient walking gait. Finally this is experimentally validated.

Due to the reasons detailed in Appendix C, the control-oriented model developed in Chapter III is incapable of accurately estimating the specific cost of mechanical transport. To better estimate the c_{mt} and reduce the above mentioned discrepancy, the complex model presented in Appendix C is used and a $c_{mt} = 0.18$ was estimated for the nominal fixed point for walking. The experimentally obtained c_{mt} for the nominal walking experiment is within 10% of this value. Thus, the complex model improved the discrepancy of the model predicted value and the experimentally obtained value of c_{mt} significantly.

Having identified the source of the discrepancy, we next try to use the complex model for improving the energy efficiency further. As presented in Section 4.5, one way to design energy efficient gaits is to carry out the optimization procedure for finding stable periodic walking gaits by directly minimizing c_{mt} . However, the optimization procedure only

produces a local minima. Further carrying out the optimization on the high DOF complex model is computationally infeasible. Instead, we perform parameter sweeps of various control parameters on the nominal walking fixed point and carry out simulations on the complex model to try and identify parameters that would likely effect c_{mt} significantly. These simulations are extremely slow and are run on a grid computer for faster throughput. Here we present the result of two such parameter sweeps that influence c_{mt} .

The virtual constraints for walking are illustrated in Figure 4.2. The stance injection to stance decompression transition is fixed at 70% of the gait. A parameter sweep of the location of this transition was carried out on the nominal walking fixed point on the complex model. The stance injection to stance decompression transition was varied in the range of $[-12\%, 9\%]$ around the nominal 70% of the gait. The resulting c_{mt} values are depicted in Figure D.1, which clearly shows that changing the location of the stance injection to stance decompression influences c_{mt} . However, with no change to the nominal location of the transition (0% change) the c_{mt} is very close to a minima. Thus, although changing the location of the stance injection to stance decompression transition does change energy efficiency, it can not improve the energy efficiency beyond what we already have.

Next, an offset is introduced on the torso virtual constraint. This has the effect of either leaning the heavy torso forward or backward depending on the offset being negative or positive respectively. Leaning the torso forward moves the COM forward and speeds up the robot, and similarly leaning the torso backward slows down the robot. A negative torso offset is also applied to the swing leg angle virtual constraint so as to maintain the same range of θ for the fixed point. With this in place, a parameter sweep of the torso offset is carried in the range $[-5^\circ, 6.5^\circ]$. Figure D.2 illustrates the effect of torso offset on c_{mt} . As is evident, torso offset has a significant effect on c_{mt} . Further, leaning the torso backwards, or slowing down the robot, appears to reduce c_{mt} until torso offset reaches 6.5° .

Figure D.2 clearly illustrates that leaning the torso backward could potentially improve energy efficiency. To immediately validate this in experiment, an existing slow walking experiment was picked up where the torso offset was swept from 0° to 16° to lean the torso backwards and slow down the walking speed. Individual consecutive steps of the walking experiment are binned by the torso offset and the energy efficiency computed for each bin.

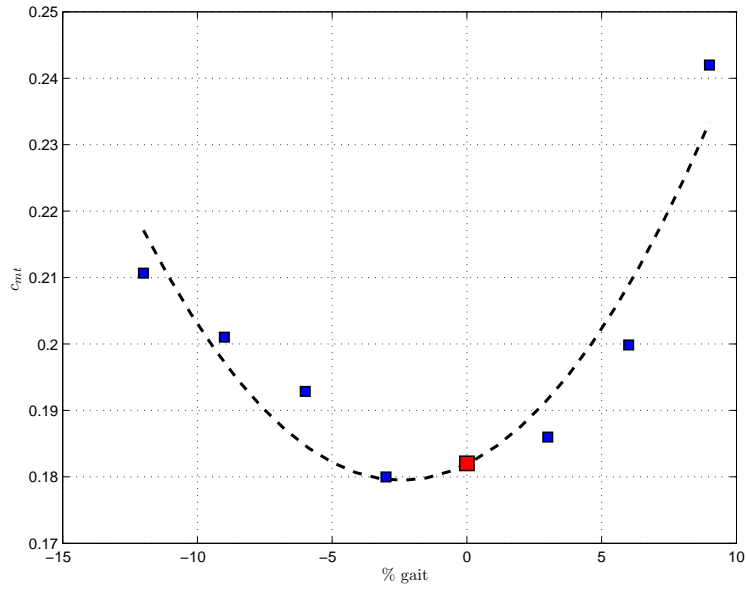


Figure D.1: Effect of location of stance injection to stance decompression transition on specific cost of mechanical transport. This is obtained by carrying out a parameter sweep and simulating the complex model on a grid computer. The thick red square indicates the value of c_{mt} at the nominal location of stance injection to stance decompression transition. The nominal transition is at 70% into the gait, and the indicated offset is applied. The dashed line indicates a second order polynomial fit of the data. The c_{mt} has a minima very close to the nominal location of the transition.

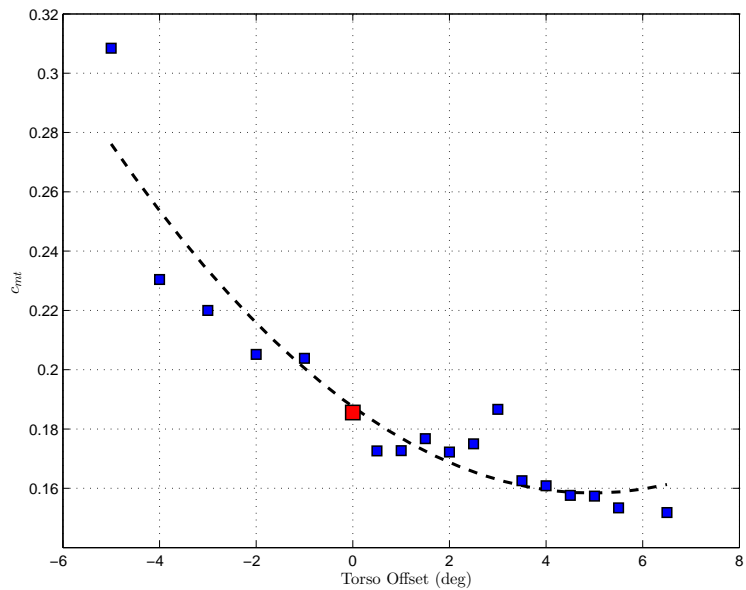


Figure D.2: Effect of torso offset on c_{mt} . This is obtained by carrying out a parameter sweep and simulating the complex model on a grid computer. A positive torso offset leans the torso backwards and decreases the walking speed, while a negative torso offset leans the torso forwards and increases the walking speed. The thick red square illustrates the value of c_{mt} for the nominal fixed point with no changes to the torso virtual constraint. The dashed line is a second order polynomial fit of the data.

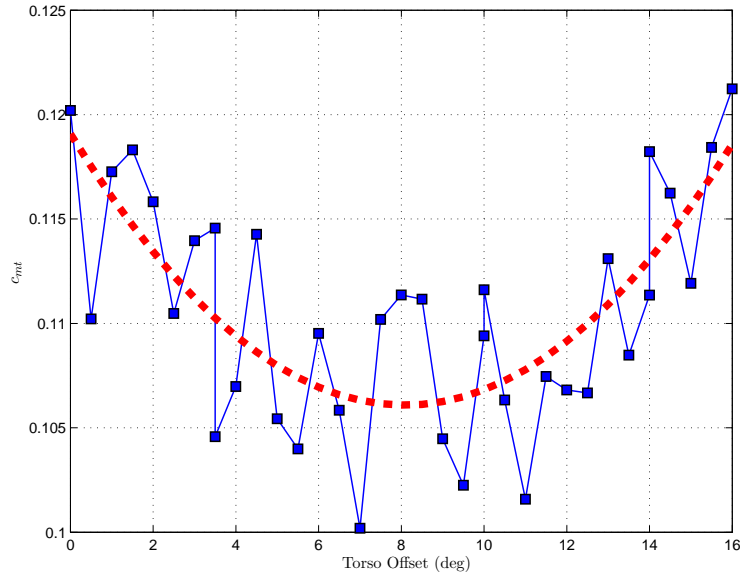


Figure D.3: Specific cost of mechanical transport computed for various values of torso offset from a slow walking experiment. A slow walking experiment is picked up and consecutive walking steps with the same torso offset are used to compute the c_{mt} for that particular torso offset. Positive torso offset values lean the torso backward, slowing down the walking speed. The number of steps for each torso offset value vary from 4 steps to 10 steps, and thus the obtained c_{mt} is only a rough estimate obtained from the relatively small sample set. The thick red dashed line is a second order polynomial fit of the data. At a torso offset of 8° , a $c_{mt} = 0.106$ is obtained from the polynomial fit. Note that the fixed point used for this experiment is neither the nominal nor the energy efficient fixed point.

The number of steps for each torso offset vary from 4 to 10 steps. Since these form a fairly small sample set, the estimated c_{mt} for each torso offset value is only a rough estimate. It is to be noted that this slow walking experiment employs a walking fixed point that is neither the nominal nor the energy efficient fixed point presented in Section 4.5. Figure D.3 plots the experimentally obtained c_{mt} values for each torso offset from this slow walking experiment. From the polynomial fit of the data points, a minima of $c_{mt} = 0.106$ is obtained when the torso offset is 8° .

BIBLIOGRAPHY

BIBLIOGRAPHY

- [1] M. Ahmadi, “Stable control of a one-legged robot exploiting passive dynamics,” Ph.D. dissertation, McGill University, Montreal, Canada, May 1998.
- [2] M. Ahmadi and M. Buehler, “Controlled passive dynamic running experiments with the arl-monopod ii,” *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 974–986, October 2006.
- [3] R. Alexander, “Three uses for springs in legged locomotion,” *The International Journal of Robotics Research*, vol. 9, no. 2, pp. 53–61, 1990.
- [4] R. M. Alexander, “Simple models for walking and jumping,” *Human Movement Science*, vol. 11, no. 1-2, pp. 3–9, February 1992.
- [5] A. Alexandrescu, *Modern C++ Design: Generic Programming and Design Patterns Applied*. Boston, MA: Addison-Wesley, 2001.
- [6] A. D. Ames, R. D. Gregg, E. D. B. Wendel, and S. Sastry, “Towards the geometric reduction of controlled three-dimensional bipedal robotic walkers,” in *3rd Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*, July 2006, pp. 183–196.
- [7] B. Andrews, B. Miller, J. Schmitt, and J. E. Clark, “Running over unknown rough terrain with a one-legged planar robot,” *Bioinspiration & Biomimetics*, vol. 6, no. 2, pp. 1–15, June 2011.
- [8] M. M. Ankarali and U. Saranlı, “Stride-to-stride energy regulation for robust self-stability of a torque-actuated dissipative spring-mass hopper,” *Chaos*, vol. 20, pp. 033 121–1 – 033 121–13, September 2010.
- [9] R. Blicckhan and R. J. Full, “Similarity in multi-legged locomotion: bouncing like a monopode,” *Journal of Comparative Physiology A*, vol. 173, pp. 509–517, 1993.
- [10] B. Brown and G. Zeglin, “The bow leg hopping robot,” in *IEEE International Conference on Robotics and Automation*, 1998.
- [11] O. Bruneau and F. B. Ouezdou, “Distributed ground/walking robot interaction,” *Robotica*, vol. 17, no. 3, pp. 313–323, 1999.
- [12] —, “Compliant contact of walking robot feet,” in *Proc. of third ECPD International Conference on Advanced Robotics, Intelligent Automation and Active Systems*, Bremen, Germany, September 1997.

- [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, March 1995.
- [14] G. A. Cavagna, N. C. Heglund, and C. R. Taylor, "Mechanical work in terrestrial locomotion: two basic mechanisms for minimizing energy expenditure," *American Journal of Physiology*, vol. 235, pp. R243–61, 1977.
- [15] G. A. Cavagna, H. Thys, and A. Zamboni, "The sources of external work in level walking and running," *Journal of Physiology*, vol. 262, pp. 639–657, 1976.
- [16] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. C. de Wit, and J. W. Grizzle, "Rabbit: a testbed for advanced control theory," *IEEE Control Systems Magazine*, vol. 23, no. 5, pp. 57–79, Oct. 2003.
- [17] C. Chevallereau, G. Bessonnet, G. Abba, and Y. Aoustin, *Bipedal Robots: Modeling, Design, and Building Walking Robots*, ser. Control Systems, Robotics and Manufacturing. John Wiley & Sons, 2009.
- [18] C. Chevallereau, D. Djoudi, and J. W. Grizzle, "Stable bipedal walking with foot rotation through direct regulation of the zero moment point," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 390–401, April 2008.
- [19] C. Chevallereau, J. W. Grizzle, and C.-L. Shih, "Asymptotically stable walking of a five-link underactuated 3d bipedal robot," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 37–50, February 2009.
- [20] ———, "Steering of a 3d bipedal robot with an underactuated ankle," in *IEEE/RSJ International Conference on Intelligent Robotics and Systems*, Taipei, Taiwan, October 2010, pp. 1242–1247.
- [21] C. Chevallereau, E. R. Westervelt, and J. W. Grizzle, "Asymptotically stable running for a five-link, four-actuator, planar bipedal robot," *The International Journal of Robotics Research*, vol. 24, no. 6, pp. 431–464, June 2005.
- [22] J.-H. Choi, "Model-based control and analysis of anthropomorphic walking," Ph.D. dissertation, University of Michigan, 2005.
- [23] J. H. Choi and J. W. Grizzle, "Feedback control of an underactuated planar bipedal robot with impulsive foot action," *Robotica*, vol. 23, no. 5, pp. 567–580, September 2005.
- [24] ———, "Planar bipedal walking with foot rotation," in *American Control Conference*, Portland, Oregon, 2005, pp. 4909–4916.
- [25] M. J. Coleman and A. Ruina, "An uncontrolled walking toy that cannot stand still," *Physical Review Letters*, vol. 80, no. 16, pp. 3658–61, Apr. 1998.
- [26] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, pp. 1082–1085, 2005.
- [27] S. H. Collins and A. Ruina, "A bipedal walking robot with efficient and human-like gait," in *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005, pp. 1983–1988.

- [28] 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, October 2006.
- [29] 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,” *The Journal of Experimental Biology*, vol. 209, pp. 171–187, 2006. [Online]. Available: <http://jeb.biologists.org/cgi/content/full/209/1/171/DC1>
- [30] V. Duindam and S. Stramigioli, “Port-based control of a compass-gait bipedal robot,” in *IFAC World Congress*, Prague, July 2005.
- [31] D. P. Ferris and C. T. Farley, “Interaction of leg stiffness and surface stiffness during human hopping,” *Journal of applied physiology*, vol. 82, pp. 15–22, 1997.
- [32] D. P. Ferris, M. Louie, and C. T. Farley, “Running in the real world: adjusting stiffness for different surfaces,” *Proceedings of The Royal Society*, vol. 265, no. 1400, pp. 989–994, June 1998.
- [33] P. S. Freeman and D. E. Orin, “Efficient dynamic simulation of a quadruped using a decoupled tree-structure approach,” *The International Journal of Robotics Research*, vol. 10, no. 6, pp. 619–627, 1991.
- [34] R. J. Full and D. E. Koditschek, “Templates and anchors: Neuromechanical hypotheses of legged locomotion on land,” *The Journal of Experimental Biology*, vol. 202, no. 23, pp. 3325–3332, November 1999.
- [35] M. Garcia, A. Chatterjee, A. Ruina, and M. Coleman, “The simplest walking model: Stability, complexity, and scaling,” *ASME Journal of Biomechanical Engineering*, vol. 120, no. 2, pp. 281–288, March 1998.
- [36] T. Geng, B. Porr, and F. Wörgötter, “Fast biped walking with a sensor-driven neuronal controller and real-time online learning,” *The International Journal of Robotics Research*, vol. 25, no. 3, pp. 243–259, 2006.
- [37] L. Geppert, “QRIO, the robot that could,” in *IEEE Spectrum*, vol. 41, 2004, pp. 34–37.
- [38] H. Geyer, A. Seyfarth, and R. Blickhan, “Compliant leg behaviour explains basic dynamics of walking and running,” *Proceedings of The Royal Society*, vol. 273, pp. 2861–2867, August 2006.
- [39] M. Gomes and A. Ruina, “Walking model with no energy cost,” *Physical Review E*, vol. 83, no. 3, pp. 032 901–1 – 032 901–4, March 2011.
- [40] A. Goswami, B. Thuilot, and B. Espiau, “A study of the passive gait of a compass-like biped robot: Symmetry and chaos,” *International Journal of Robotics Research*, vol. 17, no. 12, pp. 1282–1301, 1998.
- [41] A. Goswami, “Postural stability of biped robots and the foot-rotation indicator (fri) point,” *The International Journal of Robotics Research*, vol. 6, no. 6, pp. 523–533, June 1999.

- [42] 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.
- [43] P. Gregorio, M. Ahmadi, and M. Buehler, “Design, control and energetics of an electrically actuated legged robot,” *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, vol. 27, no. 4, pp. 626–634, 1997.
- [44] J. W. Grizzle, “Jessy Grizzle’s publications,” <http://www.eecs.umich.edu/~grizzle/papers/robotics.html>, 2010. [Online]. Available: <http://www.eecs.umich.edu/~grizzle/papers>
- [45] ——. (2010) Dynamic leg locomotion. Youtube Channel: <http://www.youtube.com/DynamicLegLocomotion>. [Online]. Available: <http://www.youtube.com/DynamicLegLocomotion>
- [46] J. W. Grizzle, G. Abba, and F. Plestan, “Asymptotically stable walking for biped robots: Analysis via systems with impulse effects,” *IEEE Transactions on Automatic Control*, vol. 46, no. 1, pp. 51–64, Jan 2001.
- [47] J. W. Grizzle, C. Chevallereau, A. Ames, and R. Sinnet, “3d bipedal robotic walking: Models, feedback control, and open problems,” in *IFAC Symposium on Nonlinear Control Systems*, Bologna, Italy, September 2010.
- [48] J. W. Grizzle, C. Chevallereau, and C.-L. Shih, “HZD-based control of a five-link underactuated 3d bipedal robot,” in *IEEE Conference on Decision and Control*, Cancun, Mexico, December 2008.
- [49] J. W. Grizzle, J. Hurst, B. Morris, H.-W. Park, and K. Sreenath, “MABEL, a new robotic bipedal walker and runner,” in *American Control Conference*, Saint Louis, MO, USA, June 2009, pp. 2030–2036.
- [50] J. W. H. Hae-Won Park, Koushil Sreenath and J. W. Grizzle, “Identification of a bipedal robot with a compliant drivetrain: Parameter estimation for control design,” *Control Systems Magazine*, vol. 31, no. 2, pp. 63–88, April 2011.
- [51] D. G. E. Hobbelen and M. Wisse, “Ankle actuation for limit cycle walkers,” *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 709–735, June 2008.
- [52] D. Hobbelen, T. de Boer, and M. Wisse, “System overview of bipedal robots flame and tulip: tailor-made for limit cycle walking,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France, September 2008, pp. 2486–2491.
- [53] J. K. Hodgins and M. H. Raibert, “Biped gymnastics,” *The International Journal of Robotics Research*, vol. 9, no. 2, pp. 115–128, April 1990.
- [54] J. K. Holm and M. W. Spong, “Kinetic energy shaping for gait regulation of underactuated bipeds,” in *IEEE International Conference on Control Applications*, San Antonio, TX, September 2008.
- [55] P. Holmes, R. J. Full, D. Koditschek, and J. Guckenheimer, “The dynamics of legged locomotion: Models, analyses, and challenges,” *SIAM Review*, vol. 48, no. 2, pp. 207–304, 2006.

- [56] Y. Hurmuzlu, F. Génot, and B. Brogliato, “Modeling, stability and control of biped robots - a general framework,” *Automatica*, vol. 40, pp. 1647–1664, 2004.
- [57] 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, February 1994.
- [58] J. W. Hurst, “The role and implementation of compliance in legged locomotion,” Ph.D. dissertation, Carnegie Mellon University, 2008.
- [59] J. W. Hurst, J. E. Chestnutt, and A. A. Rizzi, “Design and philosophy of the bi-masc, a highly dynamic biped,” in *IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007, pp. 1863–1868.
- [60] J. W. Hurst and A. A. Rizzi, “Series compliance for an efficient running gait,” *IEEE Robotics & Automation Magazine*, vol. 15, no. 3, pp. 42–51, September 2008.
- [61] S. H. Hyon and T. Mita, “Development of a biologically inspired hopping robot - ”kenken”,” in *IEEE International Conference on Robotics and Automation*, Washington DC, U.S.A., 2002, pp. 3984–3991.
- [62] F. Iida, J. Rummel, and A. Seyfarth, “Bipedal walking and running with compliant legs,” in *IEEE International Conference on Robotics and Automation*, Roma, Italy, 10-14 April 2007, pp. 3970–3975.
- [63] —, “Bipedal walking and running with spring-like biarticular muscles,” *Journal of Biomechanics*, vol. 41, no. 3, pp. 656–667, 2008.
- [64] A. Isidori, *Nonlinear Control Systems*, 3rd ed. Berlin: Springer-Verlag, 1995.
- [65] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, “Biped walking pattern generation by using preview control of zero-moment point,” in *IEEE International Conference on Robotics & Automation*, Taipei, Taiwan, September 2003, pp. 1620–1626.
- [66] S. Kajita, K. Kaneko, and M. Morisawa, “Zmp-based biped running enhanced by toe springs,” in *IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007, pp. 3963–3969.
- [67] S. Kajita, T. Nagasaki, K. Kaneko, K. Yokoi, and K. Tanie, “A running controller of humanoid biped hrp-2lr,” in *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005, pp. 616–622.
- [68] K. Kaneko, F. Kanehiro, S. Kajita, K. Yokoyama, K. Akachi, T. Kawasaki, S. Ota, and T. Isozumi, “Design of prototype humanoid robotics platform for HRP,” in *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, 2002, pp. 2431–2436.
- [69] D. E. Koditschek and M. Buhler, “Analysis of a simplified hopping robot,” *International Journal of Robotics Research*, vol. 10, no. 6, pp. 587–605, December 1991.
- [70] D. Koepl, K. Kemper, and J. Hurst, “Force control for spring-mass walking and running,” in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Montreal, Canada, July 2010, pp. 639–644.

- [71] A. D. Kuo, “Energetics of actively powered locomotion using the simplest walking model,” *Journal of Biomechanical Engineering*, vol. 124, no. 1, pp. 113–120, February 2002.
- [72] —, “Choosing your steps carefully,” *IEEE Robotics & Automation Magazine*, vol. 14, no. 2, pp. 18–29, June 2007.
- [73] C. R. Lee and C. T. Farley, “Determinants of the center of mass trajectory in human walking and running,” *The Journal of Experimental Biology*, vol. 201, no. 21, pp. 2935–2944, 1998.
- [74] S. S. M. Lee and S. J. Piazza, “Built for speed: musculoskeletal structure and sprinting ability,” *The Journal of Experimental Biology*, vol. 212, no. 22, pp. 3700–3707, August 2009.
- [75] I. R. Manchester, U. Mettin, F. Iida, and R. Tedrake, “Stable dynamic walking over rough terrain: Theory and experiment,” in *International Symposium on Robotics Research*, 2009.
- [76] P. Manoonpong, T. Geng, T. Kulvicius, B. Porr, and F. Wörgötter, “Adaptive, fast walking in a biped robot under neuronal control and learning,” *Public Library of Science Computational Biology*, vol. 3, no. 7, pp. 1305–1320, July 2007. [Online]. Available: <http://dx.plos.org/10.1371%2Fjournal.pcbi.0030134>
- [77] T. McGeer, “Passive bipedal running,” *Proceedings of the Royal Society of London*, vol. 240, no. 1297, pp. 107–34, May 1990.
- [78] —, “Passive dynamic walking,” *International Journal of Robotics Research*, vol. 9, no. 2, pp. 62–82, Apr. 1990.
- [79] T. A. McMahon and G. C. Cheng, “The mechanics of running: How does stiffness couple with speed?” *Journal of Biomechanics*, vol. 23, no. 1, pp. 65–78, 1990.
- [80] S. Miossec and Y. Aoustin, “A simplified stability study for a biped walk with underactuated and overactuated phases,” *The International Journal of Robotics Research*, vol. 24, no. 7, pp. 537–551, 2005.
- [81] A. Morecki, *Human and Machine Locomotion*. Springer, 1997.
- [82] B. Morris and J. W. Grizzle, “A restricted Poincaré map for determining exponentially stable periodic orbits in systems with impulse effects: Application to bipedal robots,” in *IEEE Conference on Decision and Control*, Seville, Spain, December 2005, pp. 4199–206.
- [83] —, “Hybrid invariance in bipedal robots with series compliance,” in *IEEE Conference on Decision and Control*, San Diego, California, USA, December 2006.
- [84] B. J. Morris and J. W. Grizzle, “Hybrid invariant manifolds in systems with impulse effects with application to periodic locomotion in bipedal robots,” *IEEE Transactions on Automatic Control*, vol. 54, no. 8, pp. 1751 – 1764, August 2009.

- [85] B. J. Morris, E. R. Westervelt, C. Chevallereau, G. Buche, and J. W. Grizzle, *Achieving Bipedal Running with RABBIT: Six Steps Toward Infinity*, ser. Lecture Notes in Control and Information Sciences. Springer Berlin / Heidelberg, 2006, vol. 340, pp. 277–297.
- [86] K. Nagasaka, Y. Kuroki, S. Itoh, and J. Yamaguchi, “Integrated motion control for walking, jumping and running on a small bipedal entertainment robot,” in *IEEE International Conference on Robotics and Automation*, New Orleans, LA, April 2004, pp. 3189–3194.
- [87] D. Owaki, M. Koyama, S. Yamaguchi, S. Kubo, and A. Ishiguro, “A 2-d passive-dynamic-running biped with elastic elements,” *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 156–162, February 2011.
- [88] O. Pascal, “Contribution à la manipulation fine et étude de la phase d’impact,” Ph.D. dissertation, Institut national polytechnique de Grenoble, 1994. [Online]. Available: <http://cat.inist.fr/?aModele=afficheN&cpsidt=169584>
- [89] F. Plestan, J. W. Grizzle, E. R. 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, August 2003.
- [90] I. Poulakakis, “Stabilizing monopedal robot running: Reduction-by-feedback and compliant hybrid zero dynamics,” Ph.D. dissertation, The University of Michigan, 2008.
- [91] 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.
- [92] ———, “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, August 2009.
- [93] J. E. Pratt and G. A. Pratt, “Exploiting natural dynamics in the control of a planar bipedal walking robot,” in *Proceedings of the Thirty-Sixth Annual Allerton Conference on Communication, Control and Computing*, Monticello, Illinois, USA, September 1998.
- [94] M. Raibert. (2011, August) PETMAN. http://www.bostondynamics.com/robot_petman.html. Boston Dynamics. Waltham, MA.
- [95] M. H. Raibert, *Legged Robots that Balance*. Cambridge, MA: MIT Press, 1986.
- [96] M. Raibert, K. Blankespoor, G. Nelson, and R. Playter, “Bigdog, the rough-terrain quadruped robot,” in *The International Federation of Automatic Control*, Seoul, Korea, July 2008, pp. 10 822–10 825.
- [97] J. R. Reubla, S. M. OConnor, and A. D. Kuo, “Human walking and running: It’s all in the redirection,” in *Dynamic Walking Conference*, 2009.

- [98] L. Righetti and A. J. Ijspeert, “Programmable central pattern generators: an application to biped locomotion control,” in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Orlando, Florida, May 2006, pp. 1585–1590.
- [99] L. Roussel, C. Canudas-De-Wit, and A. Goswami, “Generation of energy optimal complete gait cycles for biped robots,” in *IEEE International Conference on Robotics and Automation*, vol. 3, Leuven, Belgium, May 1998, pp. 2036–2041.
- [100] L. Roussel, “Génération de trajectoires de marche optimales pour un robot bipède,” Ph.D. dissertation, Institut National Polytechnique de Grenoble, November 1998.
- [101] J. Rummel and A. Seyfarth, “Stable running with segmented legs,” *The International Journal of Robotics Research*, vol. 27, no. 8, pp. 919–934, August 2008.
- [102] T. W. Z. Russ Tedrake and H. S. Seung, “Learning to walk in 20 minutes,” in *In Proceedings of the Fourteenth Yale Workshop on Adaptive and Learning Systems*, Yale University, New Haven, CT, 2005.
- [103] 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, 2006.
- [104] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higakiand, and K. Fujimura, “The intelligent ASIMO: system overview and integration,” in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, vol. 3, 2002, pp. 2478–2483.
- [105] U. Saranli and D. Koditschek, “Template based control of hexapedal running,” in *IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, September 2003, pp. 1374–1379.
- [106] U. Saranli, W. Schwind, and D. E. Koditschek, “Toward the control of a multi-jointed monopod runner,” in *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998, pp. 2676–2682.
- [107] P. Sardain and G. Bessonnet, “Forces acting on a biped robot. center of pressure–zero moment point,” *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 34, no. 5, pp. 630–637, September 2004.
- [108] T. Schaub, M. Scheint, M. Sobotka, W. Seiberl, and M. Buss, “Effects of compliant ankles on bipedal locomotion,” in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009, pp. 2761–2766.
- [109] W. J. Schwind, “Spring loaded inverted pendulum running: A plant model,” Ph.D. dissertation, University of Michigan, 1998.
- [110] A. S. Shiriaev, L. B. Freidovich, and S. V. Gusev, “Transverse linearization for controlled mechanical systems with several passive degrees of freedom,” *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 893–906, April 2010.
- [111] A. S. Shiriaev, J. W. Perram, and C. C. de Wit, “Constructive tool for orbital stabilization of underactuated nonlinear systems: Virtual constraints approach,” *IEEE Transactions on Automatic Control*, vol. 50, no. 8, pp. 1164–1176, August 2005.

- [112] G. Song and M. Žefran, “Underactuated dynamic three-dimensional bipedal walking,” in *IEEE International Conference on Robotics and Automation*, Orlando, Florida, USA, May 2006, pp. 854–859.
- [113] M. W. Spong, “Passivity based control of the compass gait biped,” in *IFAC World Congress*, 1999.
- [114] M. W. Spong and F. Bullo, “Controlled symmetries and passive walking,” *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 1025–1031, 2005.
- [115] K. Sreenath, H.-W. Park, I. Poulakakis, and J. W. Grizzle, “Compliant hybrid zero dynamics controller for achieving stable, efficient and fast bipedal walking on MABEL,” *International Journal of Robotics Research*, vol. 30, no. 9, pp. 1170–1193, August 2011. [Online]. Available: IJRR2010.pdf
- [116] R. Tajima, D. Honda, and K. Suga, “Fast running experiments involving a humanoid robot,” in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009, pp. 1571–1576.
- [117] 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.
- [118] D. J. Todd, *Walking Machines: An Introduction to Legged Robotics*. Chapman & Hall, 1985.
- [119] S. Tzafestas, M. Raibert, and C. Tzafestas, “Robust sliding-mode control applied to a 5-link biped robot,” *Journal of Intelligent Robotic Systems*, vol. 15, no. 1, pp. 67–133, 1996.
- [120] B. Vanderborght, R. V. Ham, B. Verrelst, M. V. Damme, and D. Lefeber, “Overview of the lucy project: Dynamic stabilization of a biped powered by pneumatic artificial muscles,” *Advanced Robotics*, vol. 22, no. 10, pp. 1027–1051, 2008.
- [121] B. Vanderborght, B. Verrelst, R. V. Ham, M. V. Damme, P. Beyl, and D. Lefeber, “Development of a compliance controller to reduce energy consumption for bipedal robots,” *Autonomous Robots*, vol. 24, no. 4, pp. 419–434, May 2008.
- [122] C. L. Vaughan and M. J. O’Malley, “Froude and the contribution of naval architecture to our understanding of bipedal locomotion,” *Gait & Posture*, vol. 21, no. 3, pp. 350–362, 2005.
- [123] T. Veldhuizen, “Expression templates,” in *C++ Report*, vol. 7, no. 5, June 1995, pp. 26–31.
- [124] G. Viola, “Control of underactuated mechanical systems via passivity-based and geometric techniques,” Ph.D. dissertation, Università degli Studi di Roma “Tor Vergata”, May 2008.
- [125] M. Vukobratović and B. Borovac, “Zero-moment point—thirty five years of its life,” *International Journal of Humanoid Robotics*, vol. 1, no. 1, pp. 157–73, 2004.

- [126] M. Vukobratović, B. Borovac, D. Surla, and D. Stokic, *Biped Locomotion*. Berlin: Springer-Verlag, 1990.
- [127] Q. Wei, W. Dayawansa, and P. Krishnaprasad, “Approximation of dynamical effects due to impact on flexible bodies,” in *American Control Conference*, Baltimore, MD, June 1994, pp. 1841–1845.
- [128] Q. Wei, P. Krishnaprasad, and W. Dayawansa, “Modeling of impact on a flexible beam,” in *IEEE Conference on Decision and Control*, vol. 2, San Antonio, TX, December 1993, pp. 1377–1382.
- [129] E. R. Westervelt, “Toward a coherent framework for the control of planar biped locomotion,” Ph.D. dissertation, University of Michigan, Jun. 2003.
- [130] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*. Boca Raton, FL: Taylor & Francis/CRC Press, 2007.
- [131] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, “Zero dynamics of under-actuated planar biped walkers,” in *International Federation of Automatic Control*, Barcelona, Spain, July 2002.
- [132] 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.
- [133] E. R. Westervelt, J. W. Grizzle, and C. C. de Wit, “Switching and PI control of walking motions of planar biped walkers,” *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 308–312, February 2003.
- [134] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, “Hybrid zero dynamics of planar biped walkers,” *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 42–56, Jan 2003.
- [135] M. Wisse and R. Q. v. d. Linde, *Delft Pneumatic Bipeds*, ser. Springer Tracts in Advanced Robotics. Berlin: Springer-Verlag, 2007, vol. 34.
- [136] T. Yang, E. R. Westervelt, J. P. Schmiedeler, and R. A. Bockbrander, “Design and control of a planar bipedal robot ernie with parallel knee compliance,” *Autonomous Robots*, vol. 25, no. 4, pp. 317–330, November 2008.