Systematic Controller Design for Dynamic 3D Bipedal Robot Walking

by

Brian G. Buss

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in the University of Michigan
2015

Doctoral Committee:

Professor Jessy W. Grizzle, Co-Chair
Assistant Professor Kaveh Akbari Hamed, San Diego State University, Co-Chair
Professor Anthony M. Bloch
Professor Semyon M. Meerkov
Assistant Professor C. David Remy
Assistant Professor Shai Revzen

To my family

ACKNOWLEDGEMENTS

The work presented here would not have been possible without the contribution and support of many others. I want to express my most sincere appreciation to Prof. Jessy Grizzle for his insightful guidance, continual encouragement, and great enthusiasm. I am also grateful to the National Science Foundation for their support through a Graduate Student Research Fellowship. I wish to thank the members of my dissertation committee, Prof. Kaveh Akbari Hamed, Prof. Anthony Bloch, Prof. Semyon Meerkov, Prof. C. David Remy, and Prof. Shai Revzen, for their support and insightful feedback. The method for systematic optimization of virtual constraints originated with Prof. Akbari Hamed, and I was privileged to collaborate with him in further developing and validating it. To Shelly Feldkamp, Beth Lawson, Ann Pace, and Becky Turanski, I owe many thanks for their timely, professional assistance which was always given with a smile. Dan Opila, Koushil Sreenath, and Hae-Won Park welcomed me into the lab and made my transition to graduate school easier. I want to thank Kevin Galloway for his friendship and support, especially in the rough times. Alireza Ramezani worked through numerous issues to make xPC Target work for MARLO's real-time control, and solved many other technical problems which made this work possible. The gait initiation method used in this work is largely due to him. I extend many thanks to Jonathan Hurst and his fantastic team at Oregon State University for the design and construction of MARLO, and for tireless work performing repairs and helping us overcome hardware problems. I also want to thank Gabriel Buche for many improvements to MARLO. Brent Griffin, Xingye Da, Ross Hartley, Omar Harib, and Gabriel Buche were cheerful and willing assistants as I continually interrupted them to request help with yet another experiment. I'll long remember late (very late!) nights in the lab with Brent repairing MARLO and performing experiments while listening to ThePianoGuys. The many people who have offered their friendship to me and to my family since we came to Ann Arbor have made it a memorable time in our lives. My wonderful family—Shaunae, Andrew, Clayton, Jacob, Mom, Dad, and the others—have meant everything. I could not have done this without you.

Brian G. Buss
Ann Arbor, 2015

TABLE OF CONTENTS

DEDICATION ii

ACKNOWLEDGEMENTS iii

LIST OF FIGURES vii

LIST OF TABLES viii

ABSTRACT ix

CHAPTER

1. Introduction 1

1.1 Motivation 1
1.2 Current state of research 3
1.2.1 Zero moment point 4
1.2.2 Capturability 5
1.2.3 Hybrid zero dynamics 6
1.2.4 Limit cycle walking 7
1.2.5 Intuitive approaches 8
1.2.6 Stability and disturbance rejection 10
1.3 Contributions 12
1.4 Overview of thesis 13

2. Background 14

2.1 Hybrid systems 14
2.2 Stability analysis by the method of Poincaré 16
2.3 Virtual constraints and zero dynamics 17
2.4 Hybrid zero dynamics 20

3. Hardware Description and Mathematical Models of MARLO 23

3.1 Mechanical design 23
3.2 Sensor suite 25
3.3 Generalized coordinates 27
3.4 Stance-phase dynamics 29
3.5 Impact dynamics 30
3.6 Hybrid model 32
3.7 Compliant ground model 33

4. Achieving Sustained Walking with MARLO 36

4.1 Challenges 37
4.2 Virtual constraints for 3D walking 38
4.2.1 Choosing what to control 39
4.2.2 Choosing the desired evolutions 40
4.3 Gravity compensation for robots with compliance 41
4.4 Swing leg retraction 44
4.5 Lateral control for 3D walking 48
4.5.1 Nominal SIMBICON algorithm 48
4.5.2 Swing hip angle 49
4.5.3 Torso control 51
4.6 Experiments 52
4.6.1 Method 52
4.6.2 Results 52
4.7 Summary 54

5. Systematic Controller Selection via Optimization 56

5.1 Approaches to systematic stabilization 56
5.2 Orbital stability as an optimization problem 59
5.2.1 Problem formulation 59
5.2.2 Linearized Poincaré map for the parameterized system 60
5.2.3 Optimization problem 62
5.3 Disturbance rejection as an optimization problem 63
5.3.1 Problem formulation 63
5.3.2 Discrete-time system for disturbance rejection 65
5.3.3 Optimization problem 66
5.4 Systematic controller design using a restricted Poincaré map 68
5.4.1 Parameterized restriction dynamics 70
5.4.2 Parameterized Poincaré map 73
5.4.3 Computing the linearized restricted Poincaré map 76

5.5 Formulation via bilinear matrix inequalities 78
5.6 Solver performance comparison 80

6. Stable 3D Walking with Hybrid Zero Dynamics and Systematic Optimization 83

6.1 Virtual constraints for 3D walking 84
6.2 Gait design through optimization 86
6.3 Closed-loop design and stability analysis 89
6.3.1 Feedback linearization 90
6.3.2 PD + feedforward 90
6.3.3 Event-based update 90
6.3.4 Stability analysis 91
6.4 Optimization of controlled coordinates 92
6.4.1 Parametrization of virtual constraints 93
6.4.2 Results 94
6.5 Experimental evaluation 95
6.5.1 Method 95
6.5.2 Results 98
6.5.3 Discussion 104

7. Conclusion 111

7.1 Summary of contributions 112
7.2 Future directions 114

APPENDICES 116

BIBLIOGRAPHY 122

LIST OF FIGURES

Figure

2.1	Intuitive picture of hybrid zero dynamics	22
3.1	MARLO, an ATRIAS 2.1 bipedal robot	24
3.2	Different types of foot used on MARLO	26
3.3	Coordinate definitions for MARLO model	27
3.4	Hybrid model for MARLO	32
4.1	Virtual constraints for swing leg retraction	45
4.2	Effect of swing leg retraction on disturbance rejection	46
4.3	Video snapshots from robust 2D walking experiment	47
4.4	Coordinate definitions for lateral balance control strategy	50
4.5	Coordinate plots from 3D walking experiments	53
4.6	Video snapshots from two 3D walking experiments	54
5.1	Spectral radius of $A(\xi)$ as a function of ξ.	58
5.2	Illustration of the disturbance rejection problem for bipedal walking	67
5.3	BMI versus NLP solver performance comparison	82
6.1	Feasibility of the walking gait	87
6.2	Phase portraits for the walking gait	88
6.3	Comparison between the models with and without springs	89
6.4	Laboratory setup for 3D walking experiments	96
6.5	Comparison between nominal and modified virtual constraints	98
6.6	Comparison between nominal and modified feedforward torques	100
6.7	Approximate walking paths for 3D experiments	101
6.8	Torso Euler angles with event-based updates enabled	102
6.9	Torso Euler angles with event-based updates disabled	103
6.10	Linearized position of the COM with event-based updates enabled	105
6.11	Linearized position of the COM with event-based updates disabled	106
6.12	Tracking of desired evolutions when using the nominal outputs	107
6.13	Tracking of desired evolutions when using the optimized outputs	108
A.1	Phase portraits for decoupled inverted pendulum model of walking	118

LIST OF TABLES

Table

6.1 Constraints for forward walking gait design 86
6.2 Summary of several walking experiments 99

ABSTRACT

Systematic Controller Design for Dynamic 3D Bipedal Robot Walking

by

Brian G. Buss

Chair: Jessy W. Grizzle

Virtual constraints and hybrid zero dynamics (HZD) have emerged as a powerful framework for controlling bipedal robot locomotion. Initially conceived as a method for gait and controller design and stability analysis for robots with a single unactuated degree of freedom, the theory was applied successfully to the rigid, planar bipedal robots Rabbit and ERNIE. Subsequent development of the theoretical framework allowed for systems with multiple degrees of underactuation, making possible its application to the compliant planar biped MABEL. Effective use of compliance helped MABEL walk blindly over rough terrain and set a running speed record for bipedal robots with knees. Research aimed at using HZD for 3D bipedal robots further demonstrated its potential, but also revealed an important difference with respect to the planar case with a single degree of underactuation: in the 3D case, the choice of virtual constraints has a deciding effect on the stability of a periodic orbit. Furthermore, it was found that making a good choice of virtual constraints to ensure stability of an orbit is a subtle and often challenging problem.

This thesis makes both experimental and theoretical contributions to the control of underactuated 3D bipedal robots. On the experimental side, we present the first realization of 3D dynamic walking using virtual constraints. The experimental success is achieved by augmenting a robust planar walking gait with a novel virtual constraint for the lateral swing hip angle. The resulting controller is tested in the laboratory on a human-scale bipedal robot (MARLO) and demonstrated to stabilize the lateral motion for unassisted 3D walking at approximately 1 m/s. MARLO is one of only two known robots to walk in 3D with stilt-like feet.

On the theoretical side, we introduce a method to systematically tune a given choice of virtual constraints in order to stabilize a periodic orbit of a hybrid system. The method is based on an analysis of the Poincaré map for a hybrid control system in closed loop with a parameterized feedback law. Under the assumption that the feedback law generates a periodic orbit that does not depend on the controller parameters, we introduce a Taylor series approximation of the linearized Poincaré map and formulate an optimization problem to choose the controller parameters to stabilize its fixed point. Importantly, this method makes possible systematic stabilization without the need to recompute the linearized Poincaré map at each iteration of the optimization. We demonstrate the optimization method to stabilize a walking gait for MARLO, and show that the optimized controller leads to improved lateral control compared to the nominal virtual constraints. We also describe several extensions of the basic method, allowing the use of a restricted Poincaré map and the incorporation of disturbance rejection metrics in the optimization. Together, these methods comprise an important contribution to the theory of HZD.

CHAPTER 1

Introduction

1.1 Motivation

The advantages of legs for moving about are abundantly manifest in nature. Insects and legged mammals navigate with apparent ease in even the most challenging terrain on earth. They can walk or run on uneven ground, jump over obstacles, make use of isolated footholds, and shrink their support base to weave up nearly vertical cliffs. Furthermore they do so with great elegance and energetic efficiency, making use of natural dynamics and compliance [1].

Although wheels can achieve greater energetic efficiency than legs [2], wheeled vehicles are confined to relatively smooth surfaces and only moderate slopes. Thus it is natural to consider legged robots for applications such as construction, home care, and disaster response [3, 4]. Bipedal robots, in particular, have received significant attention, due in part to the human factors of man-machine cooperation, and in part to the desire to create machines capable of interacting with the environment in the same way humans do [5]. While the motivations for this focus on bipedalism may be application dependent, it is intuitive that a machine *capable* of bipedal locomotion (possibly in addition to multipedalism or even wheeled locomotion) may be better equipped to operate in diverse environments than other

robots.[1]

While legged robots promise greater mobility than robots with wheels or tracks, the challenge of building machines which realize this mobility remains largely unmet. Although significant advances have been made since the first powered walking machines appeared over four decades ago [7], the performance of today's state-of-the-art robots is still far from that of their counterparts in nature. Most current bipedal robots will topple in the face of mild terrain variation or external disturbances [8]. A particular challenge is creating machines which *simultaneously* exhibit agility, robustness, and energetic efficiency [9–11]. Extreme examples of robots which excel in one area at the expense of another are PETMAN [12], which exhibits dynamic walking and robustness to external pushes at high energetic cost, and the Cornell Ranger [13], whose energetic efficiency surpassed even that of human walking, but which cannot clear a 5 cm obstacle.

Creating machines with performance comparable with humans in terms of speed, agility, robustness, and energetic efficiency will certainly require focused research in a broad spectrum of robotics-related disciplines, including actuator design, computer vision, planning, and control. In this thesis we focus on control, with a dedicated focus on *dynamic walking*.

Dynamic walking

A central problem in bipedal walking is how to avoid falling down. Humans and animals routinely rely on *future* reaction forces to compensate tipping moments [14]. In contrast, most bipedal walking robots (including ASIMO, HUBO, KHR-2, HRP-2, WABIAN, and others) try to compensate for destabilizing moments with *current* reaction forces (in order to keep the *zero moment point* (ZMP) in the interior of the support region) [15–17]. This constraint unnecessarily limits the allowable gaits, and typically results in slow, deliberate,

[1] JPL's DARPA Robotics Challenge contender, RoboSimian, is an interesting example of a robot using multiple means of locomotion to handle the complex environment of a simulated disaster [6].

flat-footed walking.

An alternative paradigm has emerged which emphasizes natural dynamics over strict local controllability. *Limit cycle walking* (LCW), as it is known, has its origins in the studies of purely passive walking machines initiated by Tad McGeer [18], and encompasses a broad class of approaches which aim for *orbital stability* of periodic motions [19]. Limit cycle walkers avoid falling much as humans do: by repeatedly placing their legs so as to *catch* themselves, then allowing their natural dynamics to carry them through the next step.

In this work we will pursue control strategies based on limit cycle walking. We will develop methods applicable to *underactuated* bipedal robots (that is, robots having fewer actuators than degrees of freedom). One reason for this focus is that by restricting our attention to motions satisfying a ZMP constraint, we would forfeit much of the agility which originally motivated our interest in legged locomotion. Another important reason for studying underactuation is that even a fully actuated 3D robot can become underactuated when walking. The source of underactuation may be planned (as when seeking to execute a human-like rolling foot motion) or unanticipated (such as when an uneven walking surface precludes three non-collinear points of contact, or when the object under the foot rolls or causes slipping [20]). By studying underactuation directly, we hope to establish effective and systematic methods for achieving truly dynamic walking and for handling unanticipated underactuation where it occurs.

1.2 Current state of research

Much of the literature on control of bipedal robots can be organized according to the principle used to ensure balance. As demonstrated by the examples seen so far, the degree of imbalance tolerated and the methods harnessed for dealing with imbalance vary widely. The earliest work focused on static crawlers, which maintain balance by moving slowly and

keeping their center of mass over the convex region of support provided by the feet [21]. At the other end of the spectrum, passive dynamic walkers are *never* statically stable; that they can achieve stable periodic walking (even without sensing or actuation) is a result of the stabilizing effects of foot impacts and open loop behaviors such as swing leg retraction [18,22]. Between these two extremes are a variety of approaches, *all* of which may contribute in part to the goal of creating machines which can operate reliably in a complex world.

1.2.1 Zero moment point

By far the most commonly used principle for ensuring balance in bipedal robots is the *zero moment point* (ZMP). Introduced implicitly in the work of Vukobratović as early as 1968, the ZMP is defined variously as the (unique) point on the ground at which "the influence of all forces acting on the mechanism can be replaced by a single force" [23] or "at which the moment of the total inertia force becomes zero" [24] (see also [17]).[2] The significance of the ZMP lies in the fact that if the ZMP remains in the interior of the convex support region provided by a rigid foot, the ground reaction forces acting on the foot will balance the tipping moment induced by inertial forces, and the foot will remain flat on the ground. On the other hand, when the ZMP reaches the boundary of the support region, the foot may begin rotating about its edge. Rotation of the stance foot leads to underactuation and complicates control. Therefore one approach to ensuring balance is to prescribe motions which are known *a priori* to maintain the ZMP within the support region. Using a ZMP-based control strategy, Kato's group at Waseda University first achieved "quasi-dynamic" 3D walking in 1983 and "dynamic" walking in 1984 with the bipeds WL-10R and WL-10RD [25].

The vast majority of 3D bipedal robots developed in the last three decades also use locomotion algorithms based on the ZMP [15–17, 25]. Often this principle is adopted in conjunction with an assumption that the robot is fully actuated except at the foot-ground

[2] It has been pointed out that the ZMP is equivalent to the center of pressure [24].

interface [23]. Thus, as long as the ZMP constraint is satisfied, arbitrary motions of the robot body can be achieved.

One limitation resulting from the ZMP constraint is that admissible walking patterns must maintain a non-trivial support region. Such a pattern may not be possible when kinematic constraints or terrain variations prevent the foot from making contact with the environment in three non-collinear points. Furthermore this constraint usually precludes walking gaits which exhibit human-like characteristics, including heel strike and toe off. However, several authors have presented methods for controlling the ZMP during part of a gait while still allowing the possibility of an underactuated phase [26, 27].

1.2.2 Capturability

Pratt and colleagues formalized the notion of *capturability* to characterize the essential property of a viable legged system: that it does not fall down [8, 28] (see also [10]). Capturability is defined in terms of the existence of control inputs which bring a legged system to a stop after a finite number of steps. *Capture regions* are places on the ground where a robot can step in order to come to a stop. It was shown that capture regions are analytically computable for several simple biped models. Analytically computed capture regions based on simple models were shown to be useful in rejecting disturbances in simulation and experiments with the 3D biped M2V2 [28]. In principle, this method is not restricted to particular classes of robot, gait, or feedback control structure, though formal analysis for complex robots is still a challenge.

The definition of capturability seems to place it close to static or ZMP-based approaches. Indeed, the experiments with M2V2 involved the robot being pushed while standing still. Capturability is used to plan a single step, after which the robot returns to its quiet standing. Others have used capture points for quasi-dynamic lateral walking [29]. But capturability can contribute to dynamic walking as well. PETMAN, for example, has demonstrated robust,

dynamic walking by adjusting the foot step placement relative to the instantaneous capture point [12].

1.2.3 Hybrid zero dynamics

To deal directly and effectively with underactuation, Grizzle, Westervelt, Chevallereau, and colleagues introduced the notions of virtual constrains and hybrid zero dynamics (HZD) [30–32]. Virtual constraints are functions of the robot state zeroed through feedback control; they provide an intuitive and flexible way to coordinate the links of an underactuated robot, and so to design walking motions. Additionally, virtual constraints define a subset of the state space (called the zero dynamics manifold) on which the virtually constrained system evolves. When the zero dynamics manifold is invariant under the instantaneous impact map as well as the continuous dynamics, it is possible to study stability of periodic orbits for the full-order model by analyzing the same orbit for a reduced-order model (called the restriction dynamics) [30, 33]. It is generally faster to check stability of a periodic orbit for the restriction dynamics than of the same orbit for the original system.

For a rigid, planar robot, the task of designing a zero dynamics which is compatible with the impact map is straightforward [32, Theorem 5.2]. The introduction of a constructive method (based on an event-based update of the virtual constrains) for ensuring hybrid invariance in systems with more than one degree of underactuation made HZD applicable to a much broader class of systems [34].

HZD was first demonstrated on the planar point-foot robots Rabbit and ERNIE [32]. Motivated by the energetic benefits of a rolling contact, Martin, Post, and Schmiedeler extended HZD for curved feet [35]. Allowing for compliance in the zero dynamics improved the robustness and energetic efficiency of walking with MABEL [36]. It was also shown that virtual compliance could be added to the zero dynamics for stable running [37]. MABEL set walking and running speed records for bipedal robots with knees.

Yang, et al. introduced a framework based on HZD which facilitates the control of *aperiodic* walking [38]. By designing specialized step-up, step-down, and recovery controllers and switching between them appropriately, Park and colleagues demonstrated walking over rough terrain [11,39]. Without any exteroception or prior knowledge of the terrain, the robot accommodated steps of different heights, including a 20 cm step-down (20% of leg length).

A recent development which promises further improvements in disturbance rejection is the ability to use nonholonomic virtual constraints [40]. This opens up the possibility of achieving velocity-based foot placement using the rigorous theoretical framework of HZD.

HZD has been used in the study of 3D locomotion as well [41–45]. One important discovery was that for systems with multiple degrees of underactuation, stability of a periodic orbits may depend on the particular choice of virtual constrains [42]. Three approaches for finding a stable orbit or stabilizing an existing orbit were presented. First, virtual constraints were modified based on intuition regarding the dynamics of bipedal walking. Though not systematic, a new set of virtual constraints was found which resulted in a stable orbit. Unfortunately, the same intuition did not stabilize periodic orbits for a similar model of a compliant 3D biped [45, 46]. The second approach involved redesigning the gait with a stability criterion included in the objective function. Finally, an event-based outer-loop controller was designed to stabilize a given orbit.

1.2.4 Limit cycle walking

Robots employing HZD for periodic motions are one example of the wider class of limit cycle walkers [19]. Limit cycle walking *embraces* underactuation, preferring to let the natural dynamics of the system govern its overall behavior. In addition to (purely) passive dynamic walkers, this class includes passive-based walkers which attempt to mimic the motion of unactuated walkers on flat ground with minimal control input. While passive dynamic walkers show extreme sensitivity to ground slope [47], Spong and Bullo showed how to

replicate the motion of a passive walker on a whole range of ground slopes [48]. Dubbed *controlled symmetries*, the method relies on potential energy shaping and thus requires full actuation.

Mechanical design plays an enormous role in most successful limit cycle walkers. For example, the Cornell biped and Denise make use of specially shaped feet to achieve lateral stability [49]. Denise features a one DOF hip which maintains the torso angle symmetric relative to the two leg angles in the sagittal plane. Compliance often plays an important role, reducing the actuator impedance to allow the robot to respond naturally to disturbances from the environment. Flame and TUlip both employ series elastic actuators for this purpose [4]. Additionally, passive ankle roll joints are fitted with compliant elements to help with lateral stabilization. Flame has demonstrated walking over an 8 mm (approximately 1.3% of leg length) step down. To our knowledge TUlip has only walked with assistance [50].

1.2.5 Intuitive approaches

A pioneer in *dynamic* legged robot locomotion, Marc Raibert jumped over the problem of walking to tackle hopping and running. He devised a control scheme which decomposed the task of hopping on one leg into three parts. In the vertical axis, a regular hopping oscillation was created and maintained by controlling the total energy. By adjusting the length of a springy leg at key times in a gait cycle, the hopper could either inject or remove energy from the hopping motion. In the horizontal axis, forward velocity was regulated by foot placement, while posture was regulated by adjusting the hip angle during the stance phase. To determine appropriate foot placement, Raibert introduced the *CG print*, defined as "the locus of points over which the center of gravity will travel during the next stance period" [51]. The desired position of the foot in the next stance phase was adjusted from the center of the CG print to compensate for velocity errors. Experiments with 2D [52] and 3D [53] hoppers validated the control decomposition. Raibert showed that the same method

extends readily to multi-legged robots using *one-leg gaits* (i.e., gaits for which only one leg is in contact with the ground at a time), and that by coordinating multiple legs to act as one (a so-called *virtual leg*), gaits such as bounding and trotting were possible [54]. While Raibert's work did not address walking, it is nevertheless significant because it demonstrates the viability of achieving legged locomotion—an apparently complex task—using relatively simple control schemes. Furthermore, some of the same ideas may be directly applicable to walking robots.

Other researchers have demonstrated the versatility of comparably simple control laws. Using a 3D spring loaded inverted pendulum model, Peuker, Maufroy, and Seyfarth [55] and Sharbafi et al. [56] compared several leg adjustment strategies. They found several simple strategies which resulted in stable running over a large range of parameters and initial conditions. General features of human and animal locomotion such as swing leg retraction [57,58] and the control of ground reaction forces to create a virtual pivot point [59] have also been analyzed using inverted pendulum models, and may prove useful for practical walking control strategies.

For robots that can directly control the torque at each joint, Pratt, Dilworth, and Pratt [60] described an intuitive control design method using "virtual forces". Given a collection of virtual (or hypothetical) components such as springs or dampers connected between the robot and the environment, *virtual model control* (VMC) computes joint torques which would balance the external forces the robot would experience due to the virtual components. Joint torques are computed using the Jacobian transpose. Virtual components offer intuitive parameters for the designer to adjust in tuning the controller performance. By adding a higher level controller to modify the parameters of the virtual components in an event-based manner, the researchers demonstrated walking on level ground and on moderate ramps with the planar robots Spring Turkey and Spring Flamingo [61].

Yin, Loken, and van de Panne established a control framework for controlling bipeds

in simulation [62]. Dubbed SIMBICON (SIMple BIped CONtrol), the strategy involves specifying a finite state machine and a reference pose for each state. Local PD control is used to drive each joint toward its reference angle in each state. Torso and swing hip angles are controlled relative to the world frame, and the desired swing hip angle is updated at each instant based on the distance between the CoM (or the midpoint between the hips) and the stance foot. This framework has been used in simulation of 2D and 3D bipeds to produce robust walking with a variety of gaits and in various directions. Coros et al. used a similar approach which combined reference *trajectories* with local joint control and virtual forces (as in VMC) to simulate various gaits and skills for a quadruped with a flexible spine [63]. Building on this modified framework, a trotting controller was implemented on the 3D quadrupedal robot StarlETH. The resulting gait was robust to terrain disturbances and external pushes [64].

The simplicity, flexibility, and apparent robustness of controllers designed with strategies like SIMBICON, VMC, or Raibert's decomposition are remarkable. However, these strategies do not provide tools for formally analyzing the stability of resulting gaits.

1.2.6 Stability and disturbance rejection

Underactuation in dynamic walking demands that special attention be paid to stability. We distinguish between periodic and aperiodic motions, and between motion planning and motion control.

For a given system and periodic orbit, stability is most often *assessed* using the Poincaré map. *Designing* stable periodic orbits is a more challenging task. Chevallereau et al. [42] integrated a stability criterion in an optimization to design a stable gait. The burden of computing the Poincaré map at each iteration was partially mitigated by employing HZD, which allows stability to be assessed in terms of the (much smaller) restricted Poincaré map. The method was demonstrated to design a gait for a rigid, five-link 3D biped model with

restricted yaw. An alternative approach was proposed by Diehl et al. [65], which introduced a smoothed approximation to the spectral radius. Here, again, gait design was formulated as an optimization problem. The objective function included the smoothed spectral radius and a regularization term to penalize the use of excessive actuator torque. It was shown that the smoothed spectral radius can be evaluated efficiently by solving a relaxed Lyapunov equation. However, the computational burden of computing the linearized Poincaré map apparently remains. The method was demonstrated for the design of an open-loop stable walking gait for a planar two-link walking mechanism.

Transverse linearization provides a means to stabilize both periodic and aperiodic motions for underactuated systems [66–68]. The linearization embodies the idea of a *moving Poincaré section*, where the transverse coordinates are the coordinates on the section. Transverse linearization has been used to stabilize walking of a compass biped on rough ground [69]. However, scaling the method to higher-order systems may be challenging; it would require solving the stabilization problem for a periodic linear system, which might involve, for example, solving a time-periodic matrix Riccati differential equation [66].

Assessing stability of aperiodic motions is less straightforward. Perhaps the most promising notion of stability for aperiodic legged locomotion is the *mean first passage time* (MFPT) [70]. This metric is based on a model of the biped and the environment as a stochastic system, where a subset of the state space is lumped into one absorbing (failed) state. The MFPT is the expected number of steps completed before the system enters the failed state. Recent work has suggested how the MFPT and cost of transport (a measure of energetic efficiency) can be combined in a cost function for controller design [71].

The notion of stochastic stability somewhat blurs the line between stability and disturbance rejection for bipedal robots. The gait sensitivity norm was introduced as a measure of disturbance rejection, but may prove useful for robust gait design as well [72]. Various other cost functions have been used to incorporate disturbance rejection into gait design [73].

1.3 Contributions

The work presented in this thesis builds on and contributes to the well-established framework of hybrid zero dynamics in the following ways:

First experimental realization of 3D walking with virtual constraints. We present the first experimental realization of 3D bipedal walking using virtual constraints. This is an important part of the validation of earlier theoretical results extending the theory of hybrid zero dynamics to 3D robots. The experimental success is achieved by augmenting a robust planar walking gait with a novel virtual constraint for the lateral swing hip angle. The resulting controller is tested in the laboratory on a human-scale bipedal robot (MARLO) which has no sensing or actuation in the foot or ankle. The controller stabilizes the lateral motion for unassisted 3D walking at 1 m/s.

In the course of this work we also performed an experimental comparison of the effect of swing leg retraction on disturbance rejection for a planar gait. Finally, our modified lateral hip constraint can be seen as a variation of the SIMBICON strategy; in this regard, we have provided the first (partial) experimental implementation of this strategy for a bipedal robot.

Systematic selection of virtual constraints. Virtual constraints and hybrid zero dynamics have proven extremely useful for gait design for underactuated walking robots; however a systematic means for stabilization of 3D gaits has remained elusive. We address this challenge by introducing a method to tune the choice of virtual constraints based on an optimization. In fact, the method is more general, applying not only to virtually constrained robotic systems, but to a larger class of periodic orbits for hybrid (and non-hybrid) systems. We analyze the linearized Poincaré map and show that orbital stability can be formulated as an optimization problem which can be solved efficiently with available software optimization packages.

We also extend the basic optimization framework in two ways: First, we show how

to incorporate into the optimization a measure of the ability of the closed-loop system to reject disturbances due to uncertainty in the switching manifold of the hybrid system. We also show how the optimization framework can be used in conjunction with the restricted Poincaré map. This latter extension provides one solution to the long-outstanding problem of systematically choosing virtual constraints for stability.

Experimental validation of systematic virtual constraint design. Finally, we demonstrate the optimization method to systematically stabilize a walking gait for MARLO. For this experimental work, MARLO is fitted with smaller, toroidal feet which better match the point-foot model. The optimized controller leads to improved lateral control compared to the nominal virtual constraints.

1.4 Overview of thesis

The remainder of this thesis is structured as follows: Chapter 2 reviews background material on hybrid models, virtual constraints, and zero dynamics. Chapter 3 provides a description of the robot that will be used in experiments and introduces two models for the robot: a hybrid model for control design and a more realistic model assuming compliant ground contact for controller evaluation. In Chapter 4 we present the the first successful 3D walking experiments and the associated control laws. We also present some experimental work with MARLO in planar mode, which helped in the development of the 3D controller. In Chapter 5 we present a systematic method for stabilizing gaits based on optimization of a linearized Poincaré map. The use of this optimization method is demonstrated in Chapter 6 to stabilize a 3D walking gait. Experimental comparisons of the optimized an nominal controllers are compared. Finally, Chapter 7 summarizes the conclusions and discusses future directions.

CHAPTER 2

Background

In this chapter we collect some of the facts that will be used throughout the thesis. The material here is standard. A good reference for the material on zero dynamics is [32]. Poincaré maps for continuous systems are discussed in detail in [74]; the extension to hybrid systems is discussed in [32, 33].

2.1 Hybrid systems

A *hybrid system* consists of a discrete set of *phases*, only one of which is "active" at any time, a set of rules for determining how the system transitions among them, and, for each phase, a continuous dynamical model. The discrete dynamics of a hybrid model with M phases is defined by a directed graph with M nodes. Here we will be concerned with hybrid models with one or two phases, and we will assume that the graph is a cycle with transitions $1 \to 1$ or $1 \to 2 \to 1$.

A *hybrid control system* is a hybrid system for which the continuous dynamics includes an external input, as in

$$\dot{x} = f(x, u), \tag{2.1}$$

where $x \in \mathcal{X}$ is the *state* of the dynamical system, $\mathcal{X}$ is the state manifold, $u \in \mathcal{U}$ is a

control input, and $\mathcal{U}$ is the set of admissible controls. Augmenting (2.1) with a feedback law $u = \Gamma(x)$ results in an *autonomous (or closed-loop) hybrid system*, with continuous dynamics of the form

$$\dot{x} = \bar{f}(x) = f(x, \Gamma(x)). \tag{2.2}$$

The continuous dynamics may also have validity conditions which, when violated, indicate a change of phase (i.e., a discrete "jump" to different discrete state). We assume that the validity conditions associated with each continuous model may be expressed as $\sigma(x) > 0$, where $\sigma\colon \mathcal{X} \to \mathbb{R}$ is a $\mathcal{C}^1$ function.[1] Under mild conditions, σ defines a codimension-1 embedded submanifold $\mathcal{S}$ of $\mathcal{X}$, called the *switching manifold*.[2]

When the state of the continuous system meets the switching manifold $\mathcal{S}$, an instantaneous change occurs, and the model enters a new continuous phase. We will call this event an *impact*. The *impact map* $\Delta_{i\to i+1}\colon \mathcal{X}_i \to \mathcal{X}_{i+1}$ transfers the pre-impact state x_i^- to the post-impact state $x_{i+1}^+ = \Delta_{i\to i+1}(x_i^-)$, where the subscripts i, $i+1$ index the discrete state of the model (using addition mod M) and the superscripts "$-$" and "$+$" indicate the instants immediately before and after the impact, respectively. After the impact event, the model evolves according to the continuous dynamics of the new state.

Collecting the data yields the hybrid system $\{\Sigma_i\}_{i=1}^M$, where

$$\Sigma_i\colon \begin{cases} \dot{x}_i = f_i(x_i, u_i), & x_i^- \notin \mathcal{S}_i \\ x_{i+1}^+ = \Delta_{i\to i+1}(x_i^-), & x_i^- \in \mathcal{S}_i. \end{cases} \tag{2.3}$$

Careful definitions of *solutions* of hybrid systems are given in [32, 76]. Intuitively, the

[1] In general, there may be multiple validity conditions (i.e., σ could be vector-valued instead of scalar-valued). The assumption that the graph representing the discrete dynamics is a cycle means that we need consider only one condition in each phase.

[2] Specifically, if $\mathcal{X}$ is a $2N$-manifold and 0 is a regular value of $\sigma\colon \mathcal{X} \to \mathbb{R}$ (i.e., if the derivative $d\sigma_x$ at x is surjective at every point $x \in \sigma^{-1}(0)$), then the preimage $\sigma^{-1}(0)$ is a dimension $(2N-1)$ embedded submanifold of $\mathcal{X}$ [75]. Any (relatively) open subset of $\sigma^{-1}(0)$ is also an embedded submanifold of $\mathcal{X}$.

solution starting from $x_0 \in \mathcal{X}_i \setminus \mathcal{S}_i$ is formed by integrating the continuous dynamics until the trajectory intersects the switching manifold $\mathcal{S}_i$, then applying the impact map $\Delta_{i\to i+1}$ and repeating this process. (If the initial condition x_0 lies in $\mathcal{S}_i$, then one first applies the impact map, then integrates.)

2.2 Stability analysis by the method of Poincaré

Consider a single-phase closed-loop hybrid system Σ of the form (2.3) (where we take $u_i = u \equiv 0$, so that we may think of Σ as being autonomous). Let $\varphi(t, x_0)$ be the solution of the continuous dynamics $\dot{x} = f(x)$ at time t starting from x_0 at time $t = 0$. Suppose there is a periodic solution $\varphi^*(t)$ with minimal period $T^* > 0$. The standard method for assessing the stability of the orbit $\mathcal{O} := \{\, \varphi^*(t) \mid 0 \le t < T^* \,\}$ is the method of Poincaré, which we describe here.

Suppose the orbit $\mathcal{O}$ intersects the switching manifold $\mathcal{S}$ transversely at a single point x^*. Then there is some open neighborhood $U \subset \mathcal{S}$ of x^* such that, for all $x_0 \in U$, the solution $\varphi(t, x_0)$ eventually returns to $\mathcal{S}$ (intersecting transversely). For convenience, we will assume that we can take $U = \mathcal{S}$. We define the *time-to-impact map* $T_I \colon \mathcal{S} \to \bar{\mathbb{R}}_+$ by

$$T_I(x) := \inf\{\, t \ge 0 \mid \varphi(t, x) \in \mathcal{S} \,\}. \tag{2.4}$$

Then the *Poincaré map* $P \colon \mathcal{S} \to \mathcal{S}$ is defined by

$$P(x) := \varphi(T_I(\Delta(x)), \Delta(x)) = (\varphi_{T_I} \circ \Delta)(x), \tag{2.5}$$

where $\varphi_{T_I}(x) := \varphi(T_I(x), x)$.

A necessary and sufficient condition for the orbit $\mathcal{O}$ to be locally exponentially stable is that all of the eigenvalues of the linear map $DP(x^*)$ lie in the interior of the unit circle [33].

If $\Lambda\colon \mathcal{S} \to \mathbb{R}^{n-1}$ is a coordinate system on $\mathcal{S}$, then the derivative $DP(x^*)$ can be expressed in the coordinates defined by Λ as

$$D\hat{P}(x^*) = D\Lambda(x^*)\left(I - \frac{f(x^*)\frac{\partial\sigma(x^*)}{\partial x}}{\frac{\partial\sigma(x^*)}{\partial x}f(x^*)}\right)\Phi(T_I(x^*), x^*)\frac{\partial\hat{\Delta}(x^*)}{\partial x} \tag{2.6}$$

where $\hat{P} := \Lambda \circ P \circ \Lambda^{-1}$, $\hat{\Delta} := \Delta \circ \Lambda^{-1}$, and Φ is the solution of the *variational equation*

$$\begin{aligned} \dot{\Phi}(t, x_0) &= \frac{\partial f(\varphi(t, x_0))}{\partial x}\Phi(t, x_0) \\ \Phi(0, x_0) &= I. \end{aligned} \tag{2.7}$$

For more details on the variational equation, see [74].

2.3 Virtual constraints and zero dynamics

Consider a mechanical system governed by the standard second-order dynamics

$$D(q)\ddot{q} + H(q, \dot{q}) = B(q)u, \tag{2.8}$$

where $q \in \mathcal{Q} \subset \mathbb{R}^N$ is a vector of generalized coordinates and $u \in \mathcal{U} \subset \mathbb{R}^m$ is a control input. Defining the state vector $x = (q, \dot{q})$, we can write (2.8) as the first-order control-affine dynamical system

$$\dot{x} = \underbrace{\begin{bmatrix} \dot{q} \\ -D(q)^{-1}H(q, \dot{q}) \end{bmatrix}}_{f(x)} + \underbrace{\begin{bmatrix} 0 \\ D(q)^{-1}B(q) \end{bmatrix}}_{g(x)} u. \tag{2.9}$$

We assume that the system is *underactuated*, i.e., that the dimension N of q is greater than the number m of independent actuators.[3]

[3]Examples of control designs based on zero dynamics for the fully actuated [77] and overactuated [78] cases are found in the literature.

An output function $h\colon \mathcal{X} \to \mathbb{R}^m$ for the control system (2.9) is said to have *uniform vector relative degree 2* at $x_0 \in \mathcal{X}$ if $L_g L_f h(x_0)$ is nonsingular and there is a neighborhood U of x_0 such that $L_g h(x) = 0$ for all $x \in U$. A function defined on $\mathcal{Q}$ can be trivially redefined on $\mathcal{X}$, so Lie derivatives and the notion of relative degree of this function are also defined in this case.

A constraint of the form

$$h(q) = 0 \tag{2.10}$$

enforced by sensing and feedback control is called a *holonomic virtual constraint*.[4] Enforcement of virtual constraints leads to a reduction in the dimension of a dynamical system. Specifying and (approximately) enforcing appropriate virtual constraints is a flexible and intuitive way to coordinate the motions of a mechanical system. We will describe a particularly convenient class of virtual constraints and its use for gait design in Section 4.2.

Let $h\colon \mathcal{Q} \to \mathbb{R}^m$ be a smooth function. Define $y = h(q)$, and suppose we wish to ensure $y \equiv 0$. Differentiating y yields

$$\begin{aligned}
\dot{y} &= \frac{\partial h}{\partial q}\dot{q} =: L_f h \\
\ddot{y} &= \frac{\partial h}{\partial q}\ddot{q} + \frac{\partial}{\partial q}\left(\frac{\partial h}{\partial q}\dot{q}\right)\dot{q} \\
&= \underbrace{\frac{\partial h}{\partial q} D^{-1} B}_{L_g L_f h}\, u + \underbrace{\frac{\partial}{\partial q}\left(\frac{\partial h}{\partial q}\dot{q}\right)\dot{q} - \frac{\partial h}{\partial q} D^{-1} H}_{L_f^2 h}\,.
\end{aligned}$$

If h has uniform vector relative degree 2 at a point $q \in \mathcal{Q}$, then the $m \times m$ matrix $\frac{\partial h}{\partial q}(q) D(q)^{-1} B$ (called the *decoupling matrix*) is non-singular in a neighborhood of q and the feedback

$$u^*(q, \dot{q}) = \left(\frac{\partial h}{\partial q} D^{-1} B\right)^{-1} \left[\frac{\partial h}{\partial q} D^{-1} H - \frac{\partial}{\partial q}\left(\frac{\partial h}{\partial q}\dot{q}\right)\dot{q}\right] \tag{2.11}$$

[4] As noted in Section 1.2, it is also possible to define and use non-holonomic virtual constraints.

results in the linear output dynamics

$$\ddot{y} = 0. \tag{2.12}$$

If the initial condition $x(t_0)$ of (2.9) is such that the output satisfies $y(t_0) = \dot{y}(t_0) = 0$, the feedback (2.11) ensures that $y(t) = \dot{y}(t) = 0 \ \forall t \geq t_0$. The maximal dynamics compatible with the output being identically zero is called the *zero dynamics*.

Hypotheses guaranteeing the existence and uniqueness of the zero dynamics for systems with a single degree of underactuation (i.e., for which $m = N - 1$) are given in [32, Lemma 5.1]; similar hypotheses assure the existence for systems with any degree of underactuation (i.e., $0 < m < N$).

Lemma 1 (Existence and uniqueness of the zero dynamics)**.** *Consider a system with dynamics* (2.8) *and an output function* $h\colon \mathcal{Q} \to \mathbb{R}^m$*. Suppose*

1. *there exists an open set* $\tilde{\mathcal{Q}} \subset \mathcal{Q}$ *such that* h *has uniform vector relative degree 2 in* $\tilde{\mathcal{Q}}$*;*
2. *there exists a smooth function* $\vartheta\colon \tilde{\mathcal{Q}} \to \mathbb{R}^{N-k}$ *such that* $\Phi = [h; \vartheta]\colon \tilde{\mathcal{Q}} \to \mathbb{R}^N$ *is an embedding; and*
3. *there exists at least one point in* $\tilde{\mathcal{Q}}$ *where* h *vanishes.*

Then the set

$$\mathcal{Z} = \left\{ (q, \dot{q}) \in T\tilde{\mathcal{Q}} \mid h(q) = 0, \ \frac{\partial h}{\partial q}\dot{q} = 0 \right\} \tag{2.13}$$

is a smooth embedded submanifold of $T\mathcal{Q}$ *of dimension* $2(N - m)$ *and the feedback* (2.11) *renders* $\mathcal{Z}$ *invariant under the continuous dynamics* (2.8)*.* $\mathcal{Z}$ *is called the* zero dynamics manifold*. The zero dynamics is given by*

$$\dot{z} = f_{zero}(z) = f(z) + g(z)u^*(z) \tag{2.14}$$

for $z \in \mathcal{Z}$*.*

Note that with $u(q,\dot{q}) = u^*(q,\dot{q})$, the origin is not asymptotically stable for the output dynamics (2.12). Exponentially stabilizing the origin renders $\mathcal{Z}$ *attractive*; this is done by adding PD feedback control as in

$$u(q,\dot{q}) = u^*(q,\dot{q}) - \left(\frac{\partial h}{\partial q} D^{-1} B\right)^{-1} \left(\frac{K_P}{\varepsilon^2} y + \frac{K_D}{\varepsilon} \dot{y}\right), \tag{2.15}$$

where $K_P > 0$ and $K_D > 0$ are diagonal gain matrices, and ε is a tuning parameter controlling the settling time.

We have shown that under reasonable hypotheses, the set of virtual constraints defined by a function $h\colon \mathcal{Q} \to \mathbb{R}^m$ induces a dynamical system having a lower dimension than the original dynamical system. We now extend this idea to the hybrid setting.

2.4 Hybrid zero dynamics

Consider a single-phase hybrid system Σ of the form (2.3), together with an output function $h\colon \mathcal{X} \to \mathbb{R}^m$ satisfying the hypotheses of Lemma 1. Suppose a feedback controller is chosen which renders $\mathcal{Z}$ attractive and invariant. If the initial state x_0 lies in $\mathcal{Z}$, the system will continue to evolve according to the zero dynamics until the trajectory reaches the switching manifold, at which point the impact map is applied. If $\Delta(\mathcal{S} \cap \mathcal{Z}) \subset \mathcal{Z}$, then the post-impact state $x^+ = \Delta(x^-)$ will lie in $\mathcal{Z}$. In this case, the zero dynamics manifold is said to be *compatible* with the impact map. As $\mathcal{Z}$ is invariant under the forward dynamics as well, we also say that $\mathcal{Z}$ is *hybrid invariant*.

In general, the zero dynamics manifold may not be hybrid invariant. Ensuring hybrid invariance is possible by an event-based modification of the virtual constraints [34, 42]. The idea is to add to h a term h_c which is updated after each impact to zero the error. This idea is formalized by the notion of a *deadbeat hybrid extension* of Σ.

We introduce a set of parameters for the purpose of achieving impact invariance of $\mathcal{Z}$. Denote these parameters by κ and suppose they take values in $\mathcal{K} \subset \mathbb{R}^r$. At each impact, the value of κ will be updated according to a rule $v\colon \mathcal{S} \to \mathcal{K}$ which will be described shortly. Adjoining κ to Σ gives the parameterized extension

$$\Sigma_e\colon \begin{cases} (\dot{x}, \dot{\kappa}) = (f(x) + g(x)u, 0)\,, & (x^-, \kappa^-) \notin \mathcal{S} \times \mathcal{K} \\ (x^+, \kappa^+) = \Big(\Delta(x^-), v(x^-)\Big)\,, & (x^-, \kappa^-) \in \mathcal{S} \times \mathcal{K}. \end{cases} \tag{2.16}$$

For convenience we define $x_e := (x, \kappa)$, $f_e(x_e) := (f(x), 0)$, $\Delta_e(x, \kappa) = (\Delta(x), v(x))$, $\mathcal{X}_e := \mathcal{X} \times \mathcal{K}$ and $\mathcal{S}_e := \mathcal{S} \times \mathcal{K}$.

We then define a sufficiently smooth extended output function $h_e\colon \mathcal{Q} \times \mathcal{K} \to \mathbb{R}^m$ as

$$h_e(q, \kappa) := h(q) - h_c(q, \kappa), \tag{2.17}$$

where $h_c(q, \kappa)$ vanishes for q such that $\theta(q) = \theta^-$. Finally, we let v be a function such that

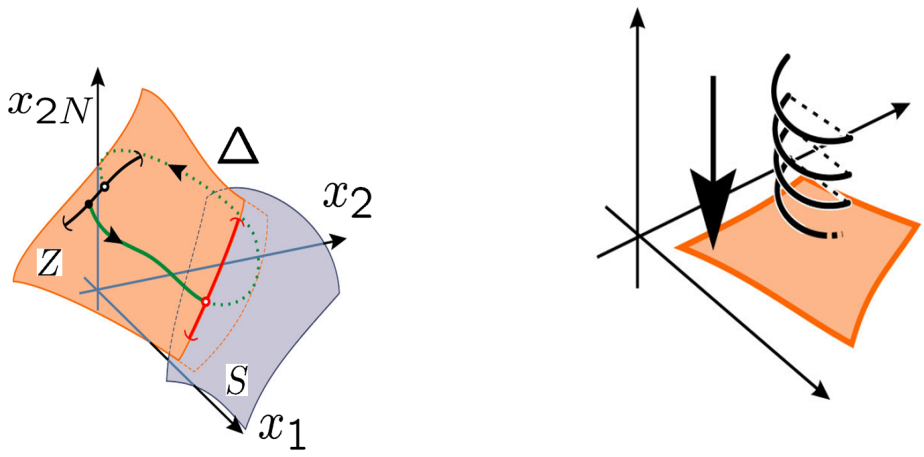
$$\begin{aligned} h_e(\Delta(x), v(x)) &= 0 \\ L_{f_e} h_e(\Delta(x), v(x)) &= 0 \end{aligned} \tag{2.18}$$

for every $(x, \kappa) \in \mathcal{S}_e \cap \mathcal{Z}_e$, where $\mathcal{Z}_e := \{\, (x, \kappa) \in \mathcal{X}_e \mid h_e(x, \kappa) = 0,\ L_{f_e} h_e(x, \kappa) = 0 \,\}$. This update law ensures that $\Delta_e(\mathcal{S}_e \cap \mathcal{Z}_e) \subset \mathcal{Z}_e$, which leads us to define the hybrid system

$$\Sigma_e|_{\mathcal{Z}_e}\colon \begin{cases} (\dot{z}, \dot{\kappa}) = (f_{zero}(z, \kappa),\ 0) & (z^-,\ \kappa^-) \notin \mathcal{S}_e \cap \mathcal{Z}_e \\ (z^+,\ \kappa^+) = \Big(\Delta(z^-),\ v(z^-)\Big) & (z^-,\ \kappa^-) \in \mathcal{S}_e \cap \mathcal{Z}_e, \end{cases} \tag{2.19}$$

called the *hybrid zero dynamics*. Figure 2.1 illustrates the properties of hybrid invariance and attractivity.

The importance of the hybrid zero dynamics is in part due to the fact that it facilitates



(a) (b)

Figure 2.1: Intuitive picture of hybrid zero dynamics. (a) The zero dynamics manifold $\mathcal{Z}$ is a dimension $2(N-m)$ submanifold of $\mathcal{X}$. Hybrid invariance of $\mathcal{Z}$ is illustrated by the green trajectory which remains in $\mathcal{Z}$ even after application of the impact map Δ. (b) Attractivity of $\mathcal{Z}$ implies that trajectories that start off the zero dynamics manifold converge to it.

stability analysis. Specifically, consider a periodic orbit $\mathcal{O}$ of the full system Σ whose closure intersects $\mathcal{S}$ transversely. Then the so-called *trivial lift* $\mathcal{O}_e := \mathcal{O} \times \{\kappa^*\}$ (where $\kappa^* := v(\bar{\mathcal{O}} \cap \mathcal{S})$) is also an orbit of the hybrid extension Σ_e. Finally, if h vanishes on $\mathcal{O}$, then $\mathcal{O}_e$ is a periodic orbit of the hybrid zero dynamics. If the transverse dynamics (i.e., the dynamics $\ddot{y} = \frac{K_P}{\varepsilon^2} y + \frac{K_D}{\varepsilon} \dot{y}$ resulting from (2.15)) converge sufficiently rapidly, then local exponential stability of $\mathcal{O}$ for the full system is equivalent to local exponential stability of $\mathcal{O}_e$ for the hybrid zero dynamics [33, 34].

CHAPTER 3

Hardware Description and Mathematical Models of MARLO

In this chapter we provide a description of MARLO, the bipedal robot used to validate the control methods developed in this thesis. We also present two mathematical models of MARLO. The first is a hybrid model used for gait and controller design. The hybrid model assumes that walking consists of alternating phases of right and left stance, with instantaneous transitions between the phases. The second incorporates a compliant ground model with friction; this model allows us to evaluate the effects of non-instantaneous double support and non-trivial feet. The robot and hybrid model were previously described in [45], where more details can be found.

3.1 Mechanical design

MARLO is one of three ATRIAS 2.1 bipedal robots designed by Jonathan Hurst at Oregon State University's Dynamic Robotics Lab. The robot has a rigid torso and two three-degree-of-freedom (DOF) legs. Each leg is composed of a four-bar linkage and a pair of brushless DC motors. The motors are connected to two of the links of the four-bar linkage through 50 : 1 harmonic drives and fiberglass leaf springs, forming a type of series elastic

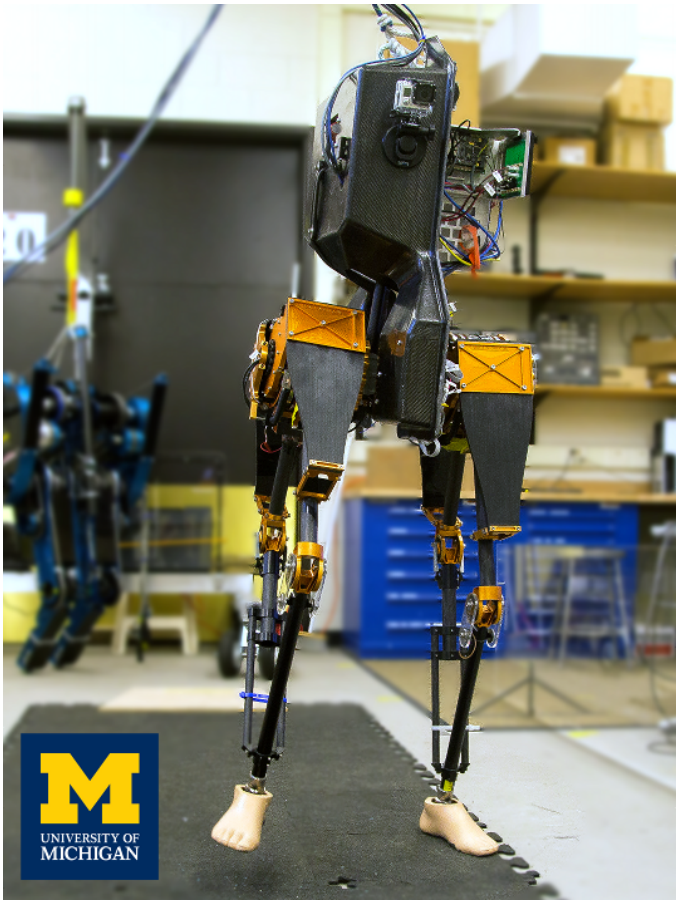


Figure 3.1: MARLO, an ATRIAS 2.1 bipedal robot. ATRIAS-series robots were designed by Jonathan Hurst and the Dynamic Robotics Laboratory at Oregon State University. (Photo: Joseph Xu)

actuator. The springs serve as energy storage elements, which may enable more efficient locomotion especially at high speeds. They also isolate the motors from impact at the feet, and provide an indirect means of estimating ground contact forces. The motors and harmonic drives are stacked into a compact hip module weighing approximately 12 kg (approximately 25% of the mass of the robot). The links comprising the leg are made of carbon-fiber tubes and account for approximately 5% of the total mass of the robot.

In addition to the two DOF of the four-bar linkge in the sagittal plane, each leg has one DOF for hip adduction and abduction. Hip adduction is driven by brushless DC motors located at the top of the torso, acting through a gear ratio of $26.7 : 1$.

The torso accounts for approximately 40% of the total mass of the robot and has room to house the on-board real-time computing, LiPo batteries, and power electronics for the motors. Off-board power was used for all experiments reported in this thesis. For the experiments reported in Chapter 4, the real-time computer was off-board; for later experiments, the real-time computer was on-board. The overall mass of the robot is 55 kg.

MARLO can be fitted with several different types of foot; these are shown in Figure 3.2. For planar walking, simple point feet are used. For the 3D experiments reported in Chapter 4, commercial prosthetic feed are used. These feet allow the robot to stand motionless without assistance before initiating a walking gait. For later experiments, a smaller (8 cm diameter) toroidal foot is used. This foot better matches the point-foot model while still providing some torque to limit yawing.

3.2 Sensor suite

MARLO is equipped with proprioceptive sensors only. Eight 32-bit absolute encoders measure the angles of the upper leg links and harmonic drive outputs relative to the torso. The nominal resolution is 5.6×10^{-7} degrees; however the noise floor is well above the quan-

Figure 3.2: Different types of foot used on MARLO. From left to right: "point foot", prosthetic foot, toroidal foot. Note the force/torque sensor mounted above the toroidal foot.

tization level, resulting in an effective measurement precision on the order of 10^{-4} degrees. Incremental encoders on the hip motors count 10,000 ticks per revolution for a resolution of 0.036°; after the hip motor to hip angle gear ratio of 26.7 : 1, the effective precision of the hip angle is 0.0013°. During the experiments reported in Section 4, MARLO was equipped with a MEMS IMU (LORD Microstrain 3DM-GX3-25) to measure the orientation of the torso. For later experiments, the MEMS IMU was replaced with a fiber optic gyro-based IMU (KVH 1750).

The torso orientation is supplied to the controller at 500 Hz (Microstrain) or 1 kHz (KVH) and converted to an Euler angle representation. Encoder measurements are supplied at the same rate as the control loop (1 kHz for the early experiments, and 2 kHz later on). Derivatives of the Euler angles and of the joint angles are estimated numerically.

During the later experiments, MARLO was also fitted with a 6-axis force/torque sensor in each ankle (visible in Figure 3.2). The sensors roughly double the moment of inertia of the legs about the leg motor axis. They also increase the leg length by 8 cm. Due to problems with the force/torque sensor interface to the real-time computer, data from these sensors

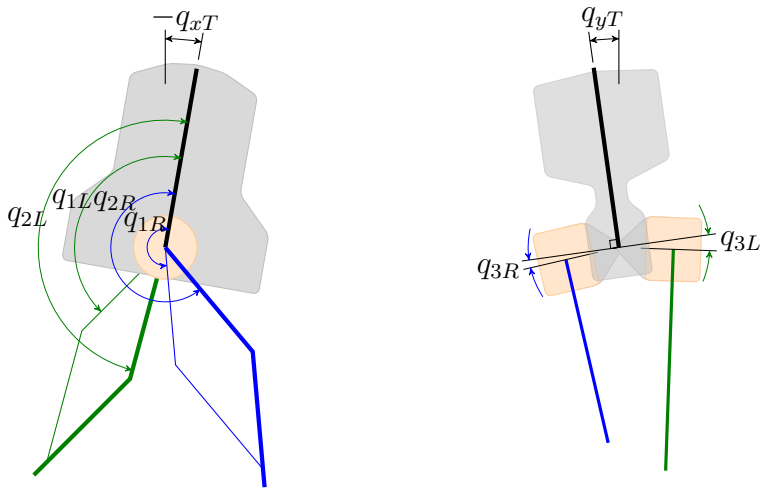


Figure 3.3: Coordinate definitions for MARLO model. Torso position and orientation are taken relative to an inertial frame. The remaining coordinates are body coordinates. In the sagittal plane, leg and gear angles are measured relative to the torso. Hip angles are also measured relative to the torso, with the angle increasing as the foot moves toward the inside.

were not used by the controller for the experiments reported here.

3.3 Generalized coordinates

To describe MARLO's orientation, we attach a coordinate frame to the torso with the z-axis pointing up and the y-axis pointing forward. The orientation R_T of MARLO's torso in the world frame is represented by the Euler angles $(q_{\mathrm{zT}}, q_{\mathrm{yT}}, q_{\mathrm{xT}})$ defined by

$$R_T = R_z(q_{\mathrm{zT}})R_y(q_{\mathrm{yT}})R_x(q_{\mathrm{xT}}),$$

where R_x, R_y, and R_z are basic rotation matrices about the x-, y-, and z-axes, respectively. We will call q_{zT}, q_{yT}, and q_{xT} the torso *yaw*, *roll*, and *pitch* angles, respectively.

All internal joint angles are defined relative to the torso. For each $i \in \{R, L\}$, the angles of the upper two links of the four-bar linkage are denoted q_{1i}, q_{2i} (shown in Figure 3.3). We also define "gear angles" that correspond to the rotation angle of the output of the harmonic

drives; these are denoted $q_{\mathrm{gr1}i}$ and $q_{\mathrm{gr2}i}$ (gear angles are not shown in Figure 3.3 for clarity). Link angles and gear angles are separated by the leaf springs. The reference gear angle is chosen so that corresponding link and gear angles are equal (e.g., $q_{\mathrm{1R}} = q_{gr1R}$) when the spring is at rest. Finally, adduction angles of the hips are defined so that they increase as the leg moves toward the mid-line; they are denoted by q_{3i}.

The absolute position of the robot in the world frame is specified by the position p_0 of the point at the base of the torso where the axes of the hip joints intersect.

For the floating base model (also called the flight-phase model) we define the vector of generalized coordinates

$$q_f = (p_0, q_{\mathrm{zT}}, q_{\mathrm{yT}}, q_{\mathrm{xT}}, q_{\mathrm{1R}}, q_{\mathrm{2R}}, q_{\mathrm{1L}}, q_{\mathrm{2L}}, q_{\mathrm{gr1R}}, q_{\mathrm{gr2R}}, q_{\mathrm{3R}}, q_{\mathrm{gr1L}}, q_{\mathrm{gr2L}}, q_{\mathrm{3L}}), \tag{3.1}$$

where the subscript f indicates flight phase. The set of possible configurations is a called the configuration manifold and is denoted $\mathcal{Q}_f$. Except near the Euler angle singularity ($q_{\mathrm{yT}} = \pm 90^\circ$), the configuration manifold $\mathcal{Q}_f$ is diffeomorphic to $\mathbb{R}^3 \times SO(3) \times \mathbb{T}^{10}$ where $\mathbb{T}^n = \mathbb{S} \times \cdots \times \mathbb{S}$ (n times) is the n-torus; however, for the present purpose we may consider $\mathcal{Q}_f$ to be an open subset of $\mathbb{R}^{N_f}$, where $N_f = 16$ is the number of degrees of freedom of the floating base model.

Generally, we will consider the robot in single support, where we assume one foot is pinned to the ground. We define the generalized coordinates in single support by removing p_0 from q_f:

$$q_s = (q_{\mathrm{zT}}, q_{\mathrm{yT}}, q_{\mathrm{xT}}, q_{\mathrm{1R}}, q_{\mathrm{2R}}, q_{\mathrm{1L}}, q_{\mathrm{2L}}, q_{\mathrm{gr1R}}, q_{\mathrm{gr2R}}, q_{\mathrm{3R}}, q_{\mathrm{gr1L}}, q_{\mathrm{gr2L}}, q_{\mathrm{3L}}). \tag{3.2}$$

We will denote the configuration manifold in single support by $\mathcal{Q}_s$, and will assume it is an open subset of $\mathbb{R}^{N_s}$, where $N_s = 13$ is the number of degrees of freedom of the model in

single support.

Finally, we will sometimes consider the model of MARLO without springs (i.e., with infinitely stiff springs). In that case the link angles $q_{1\mathrm{R}}, q_{2\mathrm{R}}, q_{1\mathrm{L}}, q_{2\mathrm{L}}$ are redundant, so we omit them. This leaves $N_f = 12$ and $N_s = 9$. For notational convenience, we will denote the coordinates of the floating base and single support models without springs by q_f and q_s; which pair of definitions we mean will be clear from the context. Thus the coordinate vectors for the model without springs are defined by

$$q_f = (p_0, q_{\mathrm{zT}}, q_{\mathrm{yT}}, q_{\mathrm{xT}}, q_{\mathrm{gr1R}}, q_{\mathrm{gr2R}}, q_{\mathrm{3R}}, q_{\mathrm{gr1L}}, q_{\mathrm{gr2L}}, q_{\mathrm{3L}}) \tag{3.3}$$

$$q_s = (q_{\mathrm{zT}}, q_{\mathrm{yT}}, q_{\mathrm{xT}}, q_{\mathrm{gr1R}}, q_{\mathrm{gr2R}}, q_{\mathrm{3R}}, q_{\mathrm{gr1L}}, q_{\mathrm{gr2L}}, q_{\mathrm{3L}}) \tag{3.4}$$

Note that, in each of the vectors (3.1)–(3.4), the last six coordinates are actuated, and the others are not.

In this chapter we distinguish between single support and flight phases with subscripts. Subsequently, we will be interested primarily with the single support model, and so, for convenience, we will omit the subscript s and write simply q or $\mathcal{Q}$.

3.4 Stance-phase dynamics

During single support we assume exactly one leg is in contact with the ground. The dynamics of the robot is that of a pinned kinematic chain. We take q_s as in (3.2) or (3.4). Using the method of Lagrange, we derive the dynamics in right stance as

$$D_R(q_s)\ddot{q}_s + C_R(q_s, \dot{q}_s)\dot{q}_s + G_R(q_s) = B(q_s)u. \tag{3.5}$$

The subscript R indicates right stance. We can express (3.5) in the form (2.1) as

$$\dot{x}_s = \underbrace{\begin{bmatrix} \dot{q}_s \\ -D_R(q_s)^{-1}(C_R(q_s,\dot{q}_s)\dot{q}_s + G_R(q_s)) \end{bmatrix}}_{f_R(x_s)} + \underbrace{\begin{bmatrix} 0 \\ D_R(q_s)^{-1}B(q_s) \end{bmatrix}}_{g_R(x_s)} u. \tag{3.6}$$

The state of the model is $x_s = (q_s, \dot{q}_s)$, and the state manifold $\mathcal{X}_s$ is the tangent bundle TQ_s. The dynamics in left stance is given by equations completely analogous to (3.5) and (3.6). Note that the state x_s is valid for both left and right stance.

This model is valid as long as the swing foot is above the ground, the normal component F_n of the ground reaction force (GRF) on the stance leg is positive, and the tangential component $F_t = (F_x, F_y)$ of the GRF is in the friction cone $\|F_t\| < \mu_s |F_n|$. When the GRF conditions fail, the robot will slip or enter flight; in the design of normal walking controllers, we do not consider solutions which exhibit these phenomena. The swing foot height condition defines the switching manifold by $\sigma_R(x_s) = p_L^z(q_s)$, where p_L^z is the vertical component of the swing foot position. When $\sigma_R(x_s)$ decreases to zero, a transition occurs from right stance to left stance.

3.5 Impact dynamics

The impact map is derived following the analysis of Hurmuzlu [79]. As the terms *swing* and *stance* are not well defined during impact, we make the convention in this case that the swing foot refers to the impacting foot. We assume that the swing leg hits the ground with non-zero vertical velocity, and that it immediately sticks (i.e., it does not slip or bounce). Simultaneously, the stance foot is assumed to lift off the ground without interaction. Furthermore, we assume that the impact duration is infinitesimal so that the configuration is constant throughout the impact event, and that the input torques u are finite.

To derive the impact map we must consider the floating base model, with q_f as in (3.1) or (3.3). As before, the method of Lagrange is applied to derive the dynamics of the floating base model, but this time we allow the possibility of external forces F_R and F_L on the end of the right and left legs. If we let $J_R = \frac{\partial p_R}{\partial q_f}$ and $J_L = \frac{\partial p_L}{\partial q_f}$ represent, respectively, the Jacobians of the right and left foot positions p_R and p_L, the dynamics of the floating base model is

$$D_f(q_f)\ddot{q}_f + C_f(q_f, \dot{q}_f)\dot{q}_f + G(q_f) = B(q_f)u + J_R^T(q_f)F_R + J_L^T(q_f)F_L. \tag{3.7}$$

By integrating the dynamics (3.7) over the infinitesimal duration of the impact, we see the post-impact state x_f^+ is related to the pre-impact state x_f^- according to

$$D_f(q_f)(\dot{q}_f^+ - \dot{q}_f^-) = J_L^T(q_f)F, \tag{3.8}$$

where F is the impact intensity (the integral of $F_L(t)$) and $F_R(t) = 0$ throughout the impact. The post-impact velocity of the impacting foot $\dot{p}_L^+ = J_L\dot{q}_f^+$ must be zero according to our assumptions; augmenting (3.8) with this constraint yields

$$\begin{bmatrix} D_f(q_f) & -J_L(q_f)^T \\ J_L(q_f) & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_f^+ \\ F \end{bmatrix} = \begin{bmatrix} D_f(q_f)\dot{q}_f^- \\ 0 \end{bmatrix}. \tag{3.9}$$

Except at the leg singularities, $D_f(q_f)$ is positive definite and $J_L(q_f)$ full rank, so the matrix on the left is non-singular. Blockwise inversion yields expressions for $\dot{q}_f^+$ and F, as

$$\begin{bmatrix} \dot{q}_f^+ \\ F \end{bmatrix} = \begin{bmatrix} D_f(q_f) & -J_L^T(q_f) \\ J_L(q_f) & 0 \end{bmatrix}^{-1} \begin{bmatrix} D_f(q_f)\dot{q}_f^- \\ 0 \end{bmatrix} = \begin{bmatrix} \bar{\Delta}_f(q_f) \\ \bar{\Delta}_F(q_f) \end{bmatrix} \dot{q}_f^- \tag{3.10}$$

where $\bar{\Delta}_F = -(J_L D_f^{-1} J_L^T)^{-1} J_L$ and $\bar{\Delta}_f = I + D_f^{-1} J_L^T \bar{\Delta}_F$. Since the stance foot is fixed in single support, the position p_0 may be computed as a function $\Upsilon(q_s)$ of the configuration.

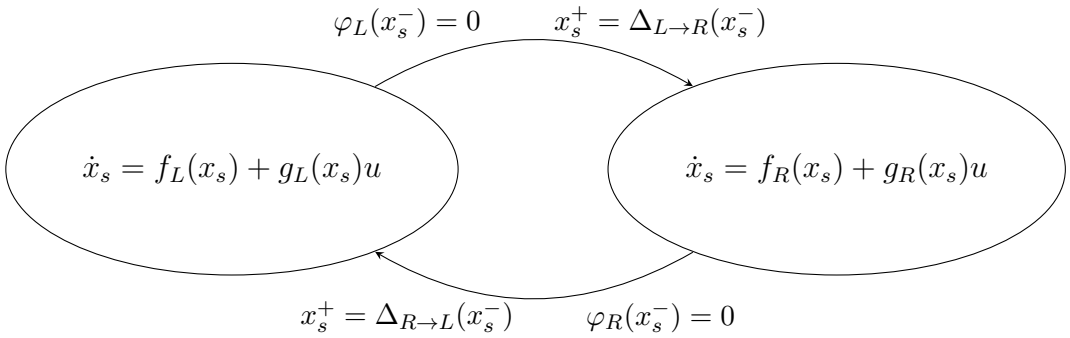


Figure 3.4: Hybrid model for MARLO. The model for 3D walking includes left and right stance phases.

Thus $\dot{q}_f = \left[\frac{\partial \Upsilon}{\partial q_s}^T \; I\right]^T \dot{q}_s$. Let $\bar{\Delta}(q_s) = [0_{N_s\times 3} \; I_{N_s\times N_s}]\, \bar{\Delta}_f(\Upsilon(q_s), q_s) \left[\frac{\partial \Upsilon}{\partial q_s}^T \; I\right]^T$; then $(q_s^+, \dot{q}_s^+) = \Delta(q_s^-, \dot{q}_s^-)$ where the impact map $\Delta_{R\to L}\colon TQ_s \to TQ_s$ is given by

$$\Delta_{R\to L}(q_s^-, \dot{q}_s^-) = \begin{bmatrix} q_s^- \\ \bar{\Delta}(q_s)\dot{q}_s^- \end{bmatrix}. \tag{3.11}$$

Note that to derive the impact map $\Delta_{L\to R}$ we need only replace $J_L(q_f)$ with $J_R(q_f)$ in (3.8)–(3.11).

3.6 Hybrid model

We assemble the continuous dynamics and impact dynamics into a hybrid model consisting of alternating phases of right and left stance, with instantaneous transition between the phases (see Figure 3.4). During each single support phase, only the stance leg is in contact with the ground. Impacts occur when the swing foot height above the ground decreases to zero. The collision is perfectly inelastic, and the former stance foot begins rising immediately.

Combining these elements yields the hybrid model

$$\Sigma_R\colon \begin{cases} \dot{x_s} = f_R(x_s) + g_R(x_s)u, & x_s^- \notin \mathcal{S}_R \\ x_s^+ = \Delta_{R\to L}(x_s^-), & x_s^- \in \mathcal{S}_R \end{cases}$$
$$\Sigma_L\colon \begin{cases} \dot{x_s} = f_L(x_s) + g_L(x_s)u, & x_s^- \notin \mathcal{S}_L \\ x_s^+ = \Delta_{L\to R}(x_s^-), & x_s^- \in \mathcal{S}_L. \end{cases} \tag{3.12}$$

We use a dynamical model of MARLO which exhibits left-right symmetry. Thus, when the desired gait is also left-right symmetric, we can consider the single-phase hybrid model

$$\Sigma\colon \begin{cases} \dot{x_s} = f_R(x_s) + g_R(x_s)u, & x_s^- \notin \mathcal{S}_R \\ x_s^+ = \Delta_{R\to R}(x_s^-), & x_s^- \in \mathcal{S}_R \end{cases} \tag{3.13}$$

where $\Delta_{R\to R} = S_x \circ \Delta_{R\to L}$, and $S_x S_x = I_{2N_s\times 2N_s}$ is a symmetry or relabeling matrix (see [32, Section 3.4.2]).

3.7 Compliant ground model

The hybrid model we have developed is useful for control design using the method of HZD described in Section 2.4. However, in practice several of the assumptions made in the development of the hybrid model are not satisfied. The most prominent discrepancy is in the foot model. While it is possible to develop a hybrid model and to apply HZD for robots with non-trivial feet [35, 77], one may not wish to do so. An alternative method for *evaluating* candidate controllers makes use of a compliant ground model. In such a model, the robot is *never* pinned to the ground; reaction forces at each foot must support the weight of the robot. A compliant ground model also makes it straightforward to model the effects of a non-instantaneous double-support phase and of slipping. Impulsive forces and torques are

not allowed in the compliant ground model. As a result, the model is continuous, not hybrid.

To model non-trivial feet, we define a collection of points $\mathcal{P}_i = \{p_{i,1}, \ldots, p_{i,n_F}\},\ i \in \{R, L\}$ rigidly attached to the end of each leg. Each foot point which lies at or below ground level is subject to reaction forces. The ground model is based on a 3D extension of a model described in the literature [80] for planar robots.[1] It defines normal reaction forces as a function of the ground penetration depth z_G and velocity $\dot{z}_G$. In the vertical direction, the model is that of a stiff, nonlinear spring-damper. The normal force is

$$F_n = k|z_G|^n - \lambda_v^a |z_G|^n \dot{z}_G - \lambda_v^b |z_G| \operatorname{sgn}(\dot{z}_G)\sqrt{|\dot{z}_G|}. \tag{3.14}$$

Tangential forces are computed with the LuGre friction model [81], which assumes the reaction force F_t is proportional to the normal force F_n. The ratio $\mu = F_t/F_n$ is a dynamic term whose derivative depends on the ground penetration depth z_G, the horizontal foot velocity v, and an internal state d representing average bristle deflection. The equation for μ is of the form

$$\bar{\mu}(d, \dot{d}, v) = \sigma_{h0} d + \sigma_{h1} \dot{d} + \alpha_{h2} v + \alpha_{h3} \operatorname{sgn}(v)\sqrt{|v|} \tag{3.15}$$

$$\mu(d, \dot{d}, v) = 0.7 \operatorname{sat}(\bar{\mu}(d, \dot{d}, v)/0.7). \tag{3.16}$$

The reaction forces in x and y directions are modeled separately, giving

$$F_x = \mu(d_x, \dot{d}_x, v_x)|F_n| \qquad \dot{d}_x = v_x - |v_x|\frac{\sigma_{h0}}{\alpha_{h0}} d_x \tag{3.17}$$

$$F_y = \mu(d_y, \dot{d}_y, v_y)|F_n| \qquad \dot{d}_y = v_y - |v_y|\frac{\sigma_{h0}}{\alpha_{h0}} d_y \tag{3.18}$$

[1] Several terms in the ground model differ slightly from the corresponding terms found in [80]. It is believed that the differences are negligible, especially when the parameters represent a very stiff ground model, as they do here.

where $F = (F_x, F_y, F_n)$ is the ground reaction force, d_x and d_y represent the friction states in the x and y directions, and $\text{sat}(x) = x$ if $|x| \leq 1$ and $\text{sat}(x) = \text{sgn}(x)$ otherwise. Note that the coefficient of friction is saturated independently in the two horizontal directions; this ensures that the reaction force lies within the *friction pyramid* defined by $|F_x| < \frac{\mu}{\sqrt{2}} F_n$ and $|F_y| < \frac{\mu}{\sqrt{2}} F_n$. The friction pyramid is a conservative linear approximation to the friction cone defined by $\sqrt{f_x^2 + f_y^2} < \mu F_n$. Both are standard in the literature [82, 83]. The friction state (d_x, d_y) is reset to zero whenever the foot point is above ground level.

When multiple foot points are in contact with the ground, a moment is induced which limits yawing. However, when only one foot point is in contact with the ground, there may be a strong tendency to yaw. To limit this effect, we include the possibility of viscous friction in the yaw axis about the end of each leg. The principle of virtual work implies that the yaw damping moment on the right leg is given by

$$M_R(q_f, \dot{q}_f) = \begin{cases} -\gamma E_R^T E_R \dot{q}_f, & \text{if } \min_{k=1,\ldots,n_F} p_{R,k}^z < 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.19}$$

where γ is the viscous friction damping constant, $p_{R,k}^z$ is the vertical component of the position of the kth right foot point and E_R is the jacobian of the last row of the rotation matrix relating the right foot frame to the body frame attached to the torso. An analogous expression gives the yaw damping moment on the left leg M_L.

The dynamics of the robot is given by

$$\begin{aligned} &D_f(q_f)\ddot{q}_f + C_f(q_f, \dot{q}_f)\dot{q}_f + G(q_f) \\ &\qquad = B(q_f)u + M_L(q_f, \dot{q}_f) + M_L(q_f, \dot{q}_f) + \sum_{k=1}^{n_F} J_{R,k}^T(q_f) F_{R,k} + \sum_{k=1}^{n_F} J_{L,k}^T(q_f) F_{L,k} \end{aligned} \tag{3.20}$$

where $F_{R,k}$ is the ground reaction force on the kth right foot point and $J_{R,k} = \frac{\partial p_{R,k}}{\partial q_f}$.

CHAPTER 4

Achieving Sustained Walking with MARLO

The theory of hybrid zero dynamics had proven extremely useful for planar walking, being used successfully on robots such as Rabbit and ERNIE [32,84]. The initial conception of the method, which required that the system have at most one unactuated degree of freedom, had been extended for models with multiple degrees of underactuation. This development allowed HZD-based control of MABEL, a planar robot with springs in series with the knee angle actuator; effective use of compliance helped MABEL walk with great energetic efficiency and even run with a strikingly human-like gait [11, 37]. The same development facilitated extension of the methods of HZD to 3D robots, and various researchers had employed it in the study of mathematical models of underactuated 3D dynamic walking [41–45]; however, to our knowledge, the only experimentally-realized sustained 3D walking to make use of virtual constraints employed a fully actuated, flat-footed gait on a NAO robot [44].

This chapter reports the first experimental demonstration of sustained dynamic 3D walking using virtual constraints. The walking was achieved with MARLO, an ATRIAS 2.1 robot designed for underactuated 3D locomotion but previously used only in a planar mode (supported laterally by a boom). The walking gait employed here was first designed for a planar model of MARLO using HZD. The initial implementation (using a modified version of the optimized gait) led to successful planar walking just three days after the first ATRIAS 2.1 robot

was fully assembled. Early efforts at implementing a systematically-designed 3D walking gait on MARLO proved challenging [46]. Therefore we used intuition and experimentation to adjust the successful planar gait and feedback controller to achieve sustained 3D walking. This success demonstrated that sustained 3D walking can be achieved on MARLO; however, it also highlights the drawbacks of "intuition." In subsequent chapters, we will address this problem with a more systematic method for stabilizing 3D walking, and will show improved lateral control in 3D walking using that method.

The rest of the chapter is organized as follows. We first outline several challenges to implementing a fully model-based gait design in these initial experiments. Section 4.2 then describes the virtual constraints used to design the gait and controller for MARLO. In Section 4.3 we describe a method for approximate gravity compensation for MARLO. Gravity compensation helped mitigate the effects of torque saturation and communication delay by improving enforcement of virtual constraints with lower feedback gains. In Section 4.4 we investigate the role of swing leg retraction in disturbance rejection during planar walking experiments. Lateral stabilization for 3D walking is addressed in Section 4.5, where we describe the intuitive idea that led to a modified choice of virtual constraints for the lateral hip angles. Section 4.6 presents experimental results.

4.1 Challenges

Being a prototype experimental robot with novel mechanical and electrical designs, MARLO presents several challenges to control. Here we briefly summarize a few that were significant influences in the experimental work reported in this chapter.

The first challenge is uncertainty in the robot dynamic model. The torso inertia is not well known due to its heterogeneous construction. More significantly, harmonic drives in series with each sagittal-plane leg motor exhibit nonlinear, position- and velocity-dependent

friction. (In fact, position-dependent stiction turned out to be an indicator of motor commutation errors associated with deteriorating sensors inside the motors.) Additionally, the point-foot model is only a rough approximation of the rigid prosthetic foot (see Figure 3.2) used in these early 3D walking experiments.

Another factor motivating experimental controller adaptations is the relatively low torque output achievable with the current power system. The continuous current rating for the amplifiers limits motor torque to 3 Nm. With an ideal transmission, this would equate to 150 Nm at each link, which is just enough to support the weight of the robot on one leg when the knee is bent 67 degrees. However, friction in the harmonic drives reduces the output torque applied to the links; the threshold of motion of the static friction is around 0.5 Nm, or 17% of the maximum motor torque. Further complicating this issue is that neither motor torques nor currents are measured; we only know the commanded current to each motor. Furthermore, we later discovered a 4–6 ms round-trip delay affecting torque commands and encoder measurements sent between the MATLAB-based controller and the firmware on low-level interface boards.

Despite these challenges, MARLO successfully walked in 3D on rigid prosthetic feet without any sensing or actuation in the foot or ankle, becoming the first human-scale bipedal robot to do so. This work lays a foundation for experimental validation of the systematic design methods we will describe in subsequent chapters.

4.2 Virtual constraints for 3D walking

Recall that a holonomic virtual constraint is a holonomic constraint enforced through the action of a feedback controller to coordinate the motion of a mechanical system. One

convenient form of virtual constraint is given by

$$0 = h(q) = h_0(q) - h_d(\theta(q)), \tag{4.1}$$

where the function $h_0 \colon \mathcal{Q} \to \mathbb{R}^m$ determines which quantities are to be controlled and $h_d \colon \mathbb{R} \to \mathbb{R}^m$ determines the desired evolution of the controlled quantities as a function of the so-called mechanical phase variable $\theta \colon \mathcal{Q} \to \mathbb{R}$.

4.2.1 Choosing what to control

The choice of controlled variables involves several considerations. First, it is convenient to choose variables which have intuitive meaning for the designer. For bipedal robots, variables such as leg and knee angles make more sense than motor shaft angles. The controlled variables must be chosen so that the output h satisfies the technical conditions described in Lemma 1 ensuring that we will be able to compute a feedback law $u = \Gamma(q, \dot{q})$ to enforce the constraint. Most importantly, the choice of controlled variables can have a significant impact on the stability of a gait [42]. This is especially relevant for 3D walking on point feet, where the unactuated dynamics in the lateral plane make stabilization particularly challenging.

Here we give one choice of controlled variables for MARLO which is intuitive and which satisfies the required technical conditions. We will find that it works well for planar gaits, but must be modified to achieve sustained 3D walking.

We specify one virtual constraint for each of MARLO's six actuators. Four of the actuators are associated with movement of the legs in the sagittal plane. In prior work with the planar bipedal robot MABEL [11, 36, 37], the angle of the torso was chosen as one of the controlled variables. This choice was particularly beneficial as MABEL had a heavy torso, so controlling its motion especially just before impact was determined to improve robustness. MARLO has a relatively light torso compared to the concentrated mass of the leg actuators

located in the hips, so the benefits of controlling the torso angle in the sagittal plane are less convincing. Instead we choose the angle of the virtual leg and the knee angle to be the controlled variables in the sagittal plane. This choice has the advantage of being symmetric with respect to the swing and stance leg, and is sufficiently intuitive to make the resulting virtual constraints meaningful to the designer. Initially, we will choose the relative hip angles as the controlled coordinates in the lateral plane. The nominal set of controlled variables is

$$h_0(q) = \begin{bmatrix} q_{\text{grLA},R} \\ q_{\text{grLA},L} \\ q_{\text{grKA},R} \\ q_{\text{grKA},L} \\ q_{\text{HA},R} \\ q_{\text{HA},L} \end{bmatrix} = \begin{bmatrix} \frac{q_{\text{gr1R}}+q_{\text{gr2R}}}{2} \\ \frac{q_{\text{gr1L}}+q_{\text{gr2L}}}{2} \\ q_{\text{gr2R}} - q_{\text{gr1R}} \\ q_{\text{gr2L}} - q_{\text{gr1L}} \\ q_{3R} \\ q_{3L} \end{bmatrix}. \tag{4.2}$$

With this choice, the virtual constraints have vector relative degree two [32].

4.2.2 Choosing the desired evolutions

The mechanical phase variable θ is defined to be the absolute angle between the virtual leg and the ground in the sagittal plane, given by

$$\theta(q) = \begin{cases} \frac{\pi}{2} - q_{xT} - \frac{q_{1R}+q_{2R}}{2}, & \text{in right stance} \\ \frac{\pi}{2} - q_{xT} - \frac{q_{1L}+q_{2L}}{2}, & \text{in left stance.} \end{cases} \tag{4.3}$$

It is convenient to normalize θ to the interval $[0, 1]$ as

$$s(\theta) = \frac{\theta - \theta^+}{\theta^- - \theta^+}, \tag{4.4}$$

where θ^+ and θ^- represent the values of θ at the beginning and end of a typical walking step.

The desired evolutions h_d are chosen to be 5th order Bézier polynomials. Letting $\alpha = (\alpha_0, \ldots, \alpha_5)$, where $\alpha_k \in \mathbb{R}^6, \quad k = 0, \ldots, 5$, denote the Bézier parameters, this yields parameterized virtual constraints of the form

$$h(q, \alpha) = h_0(q) - h_d(\theta(q), \alpha), \tag{4.5}$$

where

$$h_d(\theta, \alpha) = \sum_{k=0}^{5} \frac{5!}{k!(5-k)!} s(\theta)^k (1 - s(\theta))^{5-k} \alpha_k. \tag{4.6}$$

Using the first four constraints in a planar hybrid model of MARLO, stable periodic planar gaits were found by choosing Bézier parameters through an optimization process using `fmincon`.

4.3 Gravity compensation for robots with compliance

Given a set of virtual constraints, the task of the feedback controller is to enforce the constraints by zeroing the output $h(q)$. A feedback law which asymptotically zeroes the output and renders the zero dynamics manifold forward invariant was given in (2.15); this feedback law is a special case of input-output (or feedback) linearization [85]. However, input-output linearization is known to be sensitive to model errors. This lack of robustness stems from the fact that feedback linearization attempts to cancel the nonlinearities in the input-output dynamics. Here we derive a control law which attempts to cancel only the nonlinearities due to gravity. In doing so we improve enforcement of the virtual constraints during walking experiments while avoiding some of the potential difficulties resulting from uncertainty in the inertial parameters of the model.

Consider a fully actuated, rigid dynamic system (e.g., a robotic manipulator) with generalized coordinates q_1. A Lagrange model may be written as

$$D(q_1)\ddot{q}_1 + C(q_1, \dot{q}_1)\dot{q}_1 + G(q_1) = Bu, \tag{4.7}$$

where $G(q_1) = \frac{\partial V_{grav}}{\partial q_1}(q_1)^T$ and $V_{grav}(q_1)$ is the gravitational potential energy. As the system is fully actuated, B is square and non-singular. *Gravity compensation* is realized by letting the input u take the form $u = u_0 + u_{grav}$, where $u_{grav} = B^{-1}G$. With this input, the dynamics is reduced to $D(q_1)\ddot{q}_1 + C(q_1, \dot{q}_1)\dot{q}_1 = Bu_0$; this is the dynamics of the same system in the absence of gravity.

In an underactuated system, it generally is not possible to cancel fully the effects of gravity, as the input matrix B will not be invertible. Partial cancellation of G might be achieved by a least squares solution $u_{grav} = \arg\min_u \|Bu - G\|^2$; however, where underactuation arises from compliance in series with actuation, the resulting feedforward torque may fail to significantly alter the dynamics. To see why, it is convenient to write the model in a form proposed by Spong [86]. Spong showed that under reasonable assumptions, the dynamics of such a system can be decomposed into a "rigid" dynamics and an "actuator" dynamics with elastic coupling. We let q_2 represent the actuator angles multiplied by their gear ratios and augment (4.7) with actuator dynamics and elastic coupling terms. We make a slight generalization from the original work by allowing the dimension n of q_1 to exceed the dimension m of q_2 (which is equal to the number of actuators); we simply require that the spring deflections may be written as $B^T q_1 - q_2$ (instead of $q_1 - q_2$ as in the cited work). The inertia matrix for the rigid robot is $D(q_1)$; the rotor inertias I_{zz_k} are multiplied by the squared gear ratios m_k^2 to compute the diagonal actuator inertia matrix $J = diag\{m_1^2 I_{zz_1}, \ldots, m_m^2 I_{zz_m}\}$. Potential energy is the sum of that due to gravity, which depends only on q_1, and that due to elasticity,

which is $\frac{1}{2}k(B^T q_1 - q_2)^T(B^T q_1 - q_2)$. We have

$$D(q_1)\ddot{q}_1 + C(q_1, \dot{q}_1)\dot{q}_1 + G(q_1) + Bk(B^T q_1 - q_2) = 0 \tag{4.8a}$$

$$J\ddot{q}_2 - k(B^T q_1 - q_2) = u. \tag{4.8b}$$

With this model we see there is no hope of directly (i.e., algebraically) compensating for the gravitational term $G(q_1)$, as it is coupled to the inputs u through a double integrator. However, as Spong showed, in the limit as the spring stiffness becomes infinite, the rigid model is recovered; that is, as $k \to \infty$ the dynamics (4.8) approaches

$$(D(q_1) + BJB^T)\ddot{q}_1 + C(q_1, \dot{q}_1)\dot{q}_1 + G(q_1) = Bu \tag{4.9a}$$

$$B^T\ddot{q}_1 - \ddot{q}_2 = 0 \tag{4.9b}$$

This model has only $n - m$ degrees of underactuation, while the original has n. More importantly the large components of $G(q_1)$ lie in the range of B, so the least squares solution $u_{grav} = (B^T B)^{-1}B^T G(q_1)$ will be larger.

To understand in what sense, if any, u_{grav} compensates for the effects of gravity, we consider the system (4.8) in steady-state. Suppose first that the velocity $\dot{q}_1$ is zero and the configuration q_1 is such that $G(q_1)$ is in the range of B. Then u_{grav} is a solution to $G(q_1) = Bu$, and, for $q_2 = B^T q_1 - k^{-1}u_{grav}$, the system satisfies $(\ddot{q}_1, \ddot{q}_2) = 0$. Thus (q_1, q_2) is an equilibrium configuration corresponding to the constant input u_{grav}. It follows that u_{grav} compensates the (steady-state) effect of gravity. Note also that $u_{grav} = -k(B^T q_1 - q_2)$, so the actuator and spring torques are equal, as they must be in equilibrium. If $G(q_1)$ is not in the range of B, u_{grav} will be equal to the spring torque which minimizes the effect of gravity. We can now see that the feedforward torque effectively *establishes an equilibrium position for the gear angles* q_2 close to that which the system would have in steady-state.

We may extend this analysis to the more general model which does not make the simplifying assumptions found in [86]. This is done by adding a Lagrange multiplier λ to eliminate spring deflection (i.e., to enforce the constraint $\ddot{q}_{sp} = 0$ where the $q_{sp} = J_{sp}q$ is the spring deflection). We have

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + kJ_{sp}^T J_{sp} q = Bu + J_{sp}^T \lambda \tag{4.10a}$$

$$J_{sp}\ddot{q} = 0. \tag{4.10b}$$

Note that here $G(q) = \frac{\partial V_{grav}}{\partial q}(q)$ and $kJ_{sp}^T J_{sp} q = \frac{\partial V_{spring}}{\partial q}(q)$, where $V_{spring}(q) = \frac{1}{2} k q_{sp}^T q_{sp}$. This system can be written as

$$D(q)\ddot{q} + \Pi C(q,\dot{q})\dot{q} + \Pi G(q) + k\Pi J_{sp}^T J_{sp} q = \Pi B u \tag{4.11}$$

where the projection matrix $\Pi = I - J_{sp}^T (J_{sp} D^{-1} J_{sp}^T)^{-1} J_{sp} D^{-1}$. If we define $\bar{B} = \Pi B$ and $\bar{G} = \Pi G$, the gravity compensation torque is given by

$$u_{grav} = (\bar{B}^T \bar{B})^{-1} \bar{B}^T \bar{G}. \tag{4.12}$$

4.4 Swing leg retraction

The second enhancement which provided a significant improvement in robustness comes from observations of humans and animals. These locomotors often brake or reverse the swing leg just before impact. This behavior, termed *swing leg retraction*, has been shown to improve stability robustness in spring-mass models of running [57, 58].

We implement retraction of the swing leg by a simple modification of the virtual constraints. As defined in (4.2) and (4.5), the swing leg angle is he second element of the output $y = h(q, \alpha)$; thus we need only modify the Bézier parameters corresponding to this con-

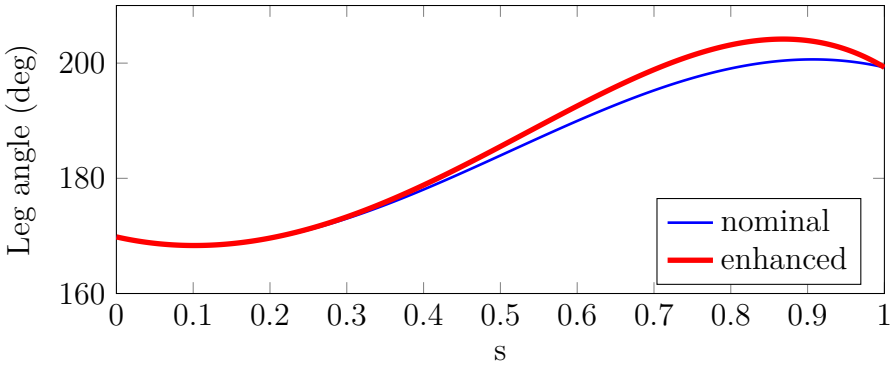


Figure 4.1: Virtual constraints for swing leg retraction.

straint. We make use of elementary properties of Bézier curves: specifically, the Mth-order Bézier curve $B(s, \alpha)$ with parameters $\alpha = (\alpha_0, \ldots, \alpha_M)$ takes the value α_0 at $s = 0$ and the value α_M at $s = 1$; furthermore, the derivative $\frac{\partial B(s,\alpha)}{\partial s}$ takes the values $M(\alpha_1 - \alpha_0)$ and $M(\alpha_M - \alpha_{M-1})$ at $s = 0$ and $s = 1$, respectively.

We do not wish to modify the step length, which is related to the swing leg angle at $s = 1$; so we leave α_M unchanged. Instead, to induce a retraction of the swing leg at impact, we increase the value of α_{M-1} for the swing leg angle virtual constrain. We could also modify the intermediate control points $\alpha_1, \ldots, \alpha_{M-2}$; however it is not usually necessary to do so.

Figure 4.1 compares the Bézier polynomials for the nominal and modified swing leg angle virtual constraints. The modified evolution was selected by adjusting α_{M-1} and running a series of walking experiments during which the boom was occasionally pushed.

Robustness to external disturbances was verified (imprecisely) by pushing or kicking the boom as MARLO walked. Adding 10 degrees to the swing leg angle component of α_{M-1} caused the horizontal velocity of the swing toe to be nearly zero at an unperturbed impact. More exaggerated leg retraction tended to cause the robot to stomp without noticeably improving stability robustness. Figure 4.2 shows the step speeds during two experiments (without and with enhanced swing leg retraction) where the boom was pushed from behind while the robot walked. Enhanced swing leg retraction allowed the robot to continue walking

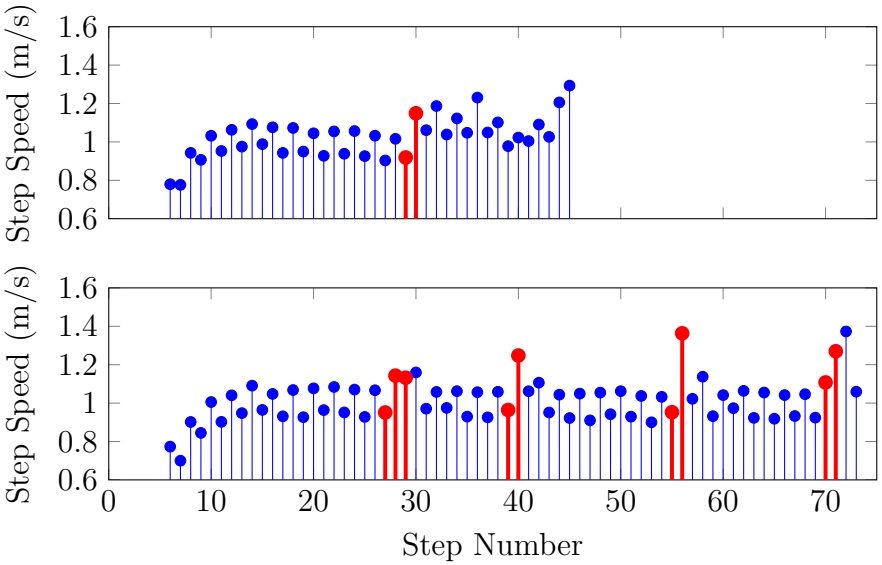


Figure 4.2: Effect of swing leg retraction on disturbance rejection. The plots indicate the step speed during two experiments where the boom was pushed. In each plot, the heavy red stems indicate steps where the experimenter was in contact with the boom. Note the significant left-right asymmetry due to the boom. (Top) Without enhanced swing leg retraction, the velocity increased after the push and remained higher than normal for multiple steps until the robot tripped and eventually fell. (Bottom) With enhanced swing leg retraction, the step velocity returns to nominal within about one step after the push. In this experiment the robot rejected multiple pushes before falling.

Figure 4.3: Video snapshots from robust 2D walking experiment. Frames are 334 ms apart. The robot took several steps after being kicked. The experimenter stopped the robot to prevent it from walking into the wall. The video is available on YouTube.

after several pushes, while it otherwise became unstable and eventually fell.

We also verified that the control design remained stable and robust when the torso pitch encoder measurement was replaced with the lower-bandwidth IMU-derived (LORD Microstrain 3DM-GX3-25) pitch angle, and when prosthetic feet were used instead of point feet. The desired knee angles were modified by hand to accommodate the prosthetic feet without scuffing. The robot successfully walked over slightly uneven terrain while subjected to external disturbances (pushes) even under these conditions. Figure 4.3 shows snapshots from a video of this experiment.

4.5 Lateral control for 3D walking

Early efforts to implement a systematically-designed 3D walking gait on MARLO were instrumental in solving new challenges related to gait initiation. However, despite some promising results, the MARLO had never taken more than about six consecutive steps [46] in 3D. The controller in use employed virtual constraints based on the controlled variables defined in (4.2). These virtual constraints give rise to a periodic orbit which is unstable [42, 45]. Instability was evident in experiments, where a common failure mode begins with poor lateral foot placement which results in the robot rolling over the stance foot. The problem is that the nominal virtual constraint makes the desired hip angle independent of torso roll instead of modulating foot placement to compensate for rolling motions, as a human would do.

To achieve lateral stability we implemented a revised hip control strategy. The essence of the strategy is to let the angles of the hips with respect to the world frame approximately mirror each other in the frontal plane. A similar "mirror law" in the sagittal plane was first proposed in [30]. Our lateral hip control law can also be seen as a modified form of the balance control strategy used in SIMBICON [62]. SIMBICON and variations thereof have been used in simulation of a variety of legged creatures [63, 87] and in experiments with a quadrupedal robot [64]. We first summarize the original algorithm, then describe the modified version used in our experiments.

4.5.1 Nominal SIMBICON algorithm

SIMBICON is a framework for the control of bipedal walking or running. It is based on a finite-state machine having a fixed target pose for each state. Within each state, PD control is used to drive individual joints toward the corresponding target angles. The swing hip and the torso angle are controlled relative to the world frame. The stance hip torque τ_A

is computed from the torso torque τ_{torso} and the swing hip torque τ_B as $\tau_A = -\tau_{torso} - \tau_B$.

One additional element is needed to provide feedback for balance. The desired swing hip angle is updated continuously by a feedback law of the form

$$\psi_{sw,d} = \psi_{sw,d0} + c_p d + c_d \dot{d} \tag{4.13}$$

where $\psi_{sw,d}$ is the instantaneous target swing hip angle, $\psi_{sw,d0}$ is the nominal target swing hip angle specified by the state machine, and d is the horizontal distance between the CoM and the stance ankle. The midpoint between the hips is used as an approximation of the CoM. In 3D, the nominal algorithm uses the same balance strategy in both the frontal and sagittal planes.

4.5.2 Swing hip angle

The experiments reported in this paper use a modified form of SIMBICON to compute the desired swing hip angle in the lateral plane. We do not use SIMBICON in the sagittal plane. We define absolute hip angles

$$\psi_R = -q_{yT} - q_{3R} \tag{4.14}$$

$$\psi_L = q_{yT} - q_{3L} \tag{4.15}$$

so that both increase as the foot moves outward. We set $\psi_{st} = \psi_R$ and $\psi_{sw} = \psi_L$ in right stance; in left stance these definitions are reversed. These definitions are illustrated in Figure 4.4.

Instead of adjusting the desired swing hip angle based on the distance d as in (4.13), we use the absolute stance hip angle ψ_{st}. This angle can be thought of as a linear approximation

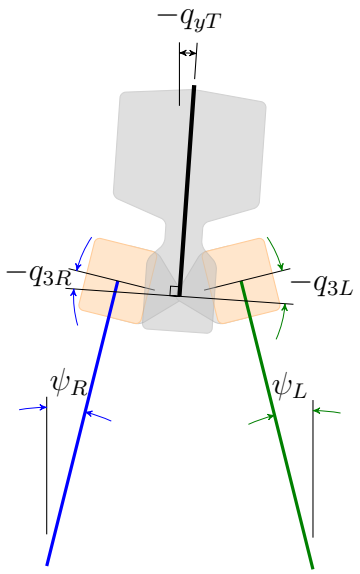


Figure 4.4: Coordinate definitions for lateral balance control strategy.

of d. The desired angle is

$$\psi_{sw,d} = \psi_{sw,d0} + c_p \psi_{st}, \tag{4.16}$$

where $\psi_{sw,d0}$ and c_p are control parameters.

One consequence of this strategy is that the swing foot generally moves *inward* during the beginning part of each step, and *outward* near the end. This is undesirable, as it brings the feet closer together during the middle of the step, increasing the likelihood that the feet will collide. It also increases tracking errors, particularly near the end of the step where they result in poor foot placement. We wish to modify (4.16) to reduce this inward motion.

For practical implementation, it is also helpful to ensure that errors near the beginning of each step are relatively small. Large post-impact errors result in large torques being applied when the support state of the robot is uncertain. Trying to correct a large error in the swing hip angle while the "swing" foot is still on the ground, for example, can generate large moments which cause the robot to yaw.

We address both of these issues simultaneously. To reduce the inward motion of the

swing foot we add a term to the right hand side of (4.16) which depends on the gait phase variable s. We also add a correction term which zeroes the error at $s = 0$ and vanishes as s approaches one. The resulting expression for the desired swing hip angle is given by

$$\psi_{sw,d} = (1-s)^3\psi_{sw} - 3(1-s)^2 s\,(b_{sw} + aq_{yT}) + \left(3(1-s)s^2 + s^3\right)(\psi_{sw,d0} + c_p\psi_{st}) \quad (4.17)$$

where $a = -1$ in right stance and $a = 1$ in left stance. The parameter b_{sw} biases the value of $\psi_{sw,d}$ in the middle of a step in order to keep the feet apart. When $s = 0$ this equation gives $\psi_{sw,d} = \psi_{sw}$, and when $s = 1$ it reduces to (4.16). Note that (4.17) defines $\psi_{sw,d}$ as cubic Bézier polynomial in s. It differs from the desired evolutions as the coefficients of the polynomial in (4.17) are updated continuously. To write the virtual constraint $0 = \psi_{sw,d} - \psi_{sw}$ in the form (4.1) we define

$$\begin{aligned} h_{0,sw}(q) = &\left(1 - (1-s)^3\right) q_{3,sw} - 3(1-s)^2 s b_{sw} \\ &+ \left(3s^2 - 2s^3\right)(a(1+c_p)q_{yT} + \psi_{sw,d0} - c_p q_{3,st})\,. \end{aligned} \quad (4.18)$$

This quantity replaces q_{3L} (in right stance) or q_{3R} (in left stance) in (4.2); the corresponding element of $h_d(\theta)$ is set to zero.

4.5.3 Torso control

Our method for controlling the torso also differs slightly from the SIMBICON strategy. Lateral torso control is easily accomplished by substituting a virtual constraint on the torso roll in place of the constraint on the stance hip. However, a satisfactory control design should also maintain the hip angles safely within their workspace. We make the tradeoff between torso and (relative) hip control explicit by defining a new actuated coordinate

$$h_{0,st}(q) = a\gamma q_{yT} + (1-\gamma)(q_{3,st} - b_{st}), \quad (4.19)$$

where b_{st} is the nominal desired stance hip angle, and $\gamma \in \mathbb{R}$. Note that $\gamma = 0$ corresponds to relative hip angle control (the nominal output function), while $\gamma = 1$ corresponds to pure torso control (as in SIMBICON). Setting $\gamma > 1$ causes the robot to lean the torso toward the stance foot, and $\gamma < 0$ causes the robot to lean the torso beyond the hip neutral position in the direction of the roll. The quantity $h_{0,st}(q)$ replaces q_{3R} (in right stance) or q_{3L} (in left stance) in (4.2); the corresponding element of $h_d(\theta)$ is set to zero.

The swing hip feedback torque is treated as a known disturbance on the torso. Its effect is canceled though disturbance feedforward.The same result is achieved in SIMBICON by the choice of τ_{stance}.

4.6 Experiments

4.6.1 Method

All 3D experiments were performed with prosthetic feet. The robot began in a static pose with the feet together. During a brief *injection phase*, the knees were extended, causing the robot to begin pitching forward. As the pitch rate crossed a predetermined threshold, a transition controller was applied. This controller uses virtual constraints similar to the walking control but with a shorter step length; it is designed to help the robot accelerate to the walking gait. After the transition step, the regular walking controller was applied. At impacts, θ^+ and α were updated to reduce post-impact errors.

4.6.2 Results

The revised lateral balance control strategy was introduced to find a baseline controller for 3D walking. Thus the initial goal in our 3D experiments was to get the robot to walk as far as possible. Whereas with the nominal virtual constraints (4.2), the robot never walked more than six steps at a time, with the revised strategy, the robot was able to walk the full

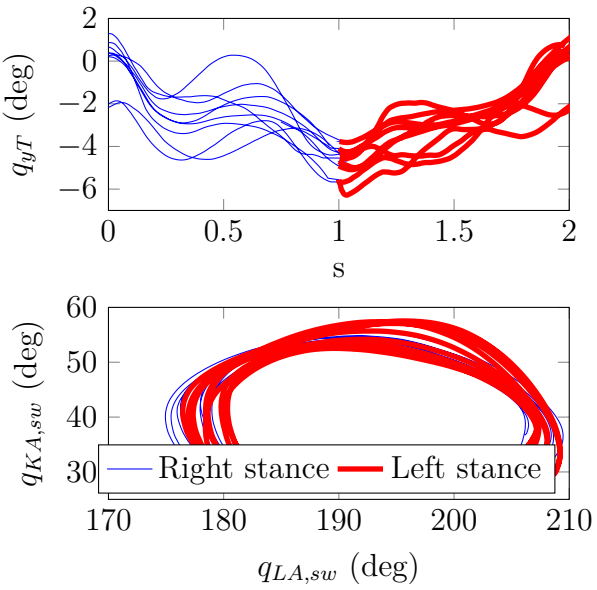


Figure 4.5: Coordinate plots from 3D walking experiments. (Top) Roll angle versus the normalized mechanical phase variable s, where s is increased by 1 in left stance to show the 2-step periodic rocking motion. (Bottom) Swing knee angle versus swing leg angle.

length of the lab repeatedly.

Proper control of torso roll facilitates lateral swing foot placement. However, when the torso was controlled without regard for the stance hip angle, there were large oscillations in both hip angles. Setting $\gamma = 0.7$ in (4.19) led to a better compromise, with increased torso movement, but reduced hip oscillations. Figure 4.5 shows the roll angle versus s and the swing leg coordinates for the middle 8 seconds of a 3D walking experiment.

With the planar controller augmented with this modified form of SIMBICON for lateral control, the robot walked both indoors on a flat laboratory floor and outdoors, on mildly sloped and uneven terrain. To achieve the outdoor walking on sloped terrain, the torso bias was manually adjusted. Outdoor walking was less robust, indicating the need for further work to enhance stability to ground variations. Snapshots from videos are shown in Fig. 4.6.

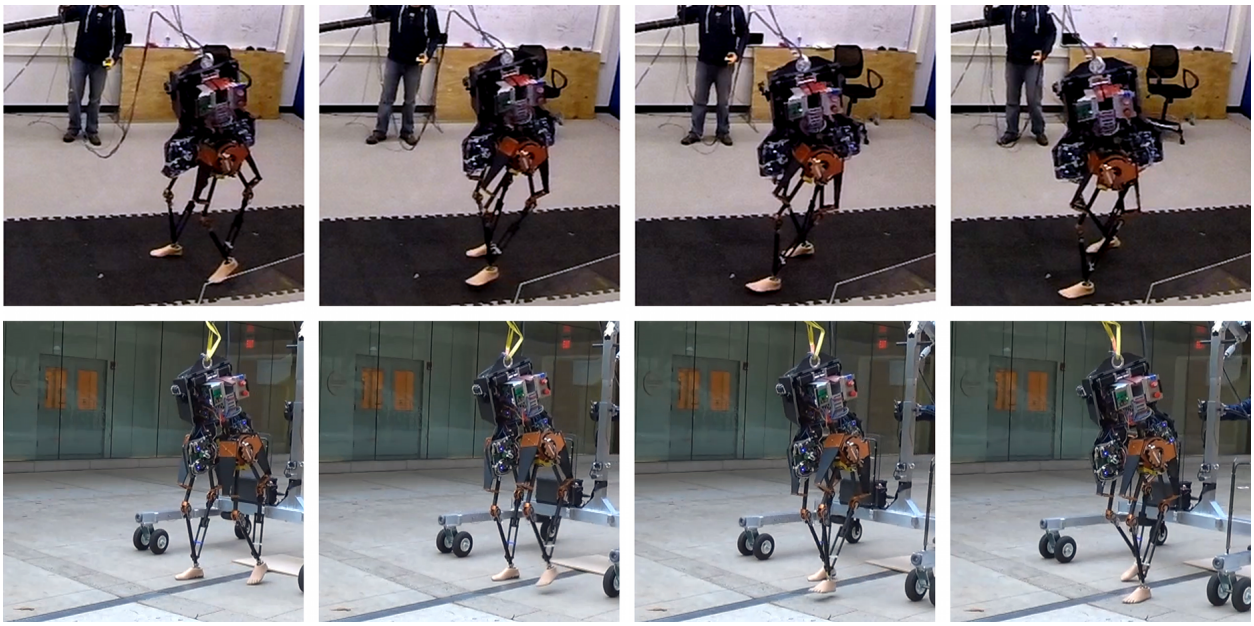


Figure 4.6: Video snapshots from two 3D walking experiments. Frames are 200 ms apart. Videos of indoor (top row) and outdoor (bottom row) experiments are available online [88].

4.7 Summary

This work demonstrated that MARLO is capable of sustained 3D walking. Starting with a planar gait, we made several modifications to improve virtual constraint enforcement and disturbance rejection. To stabilize the lateral rocking motion, we implemented novel lateral virtual constraints based on an intuition and experience. Experimental tuning of the new virtual constraints led to sustained, repeatable 3D walking in the laboratory. However, the process of finding the gait and feedback controller is unsatisfying, as it relies too heavily on intuition and experimental tuning.

Our objective is the development of general methods for gait and controller design. In the ensuing chapters, we will describe a systematic method for choosing virtual constraints to stabilize an orbit and demonstrate the method for stabilization of a 3D walking gait. The systematically-chosen virtual constraints lead to improved lateral control in 3D walking

experiments, demonstrating the utility of the method.

CHAPTER 5

Systematic Controller Selection via Optimization

A significant amount of trial-and-error preceded the successful 3D walking experiments reported in Section 4.6. This was in part due to the challenge of guiding the robot state into the basin of attraction of a periodic orbit from an initial static configuration. Better gait initiation methods would certainly help in this regard; however, gait initiation is only part of the challenge. A prior question, stated in the context of HZD, is *How can we choose virtual constraints so that a resulting periodic orbit is exponentially stable?* In this chapter we outline a method to tune virtual constraints for exponential stability based on nonlinear optimization.

Part of the material in this chapter is presented in a different form in [89, 90].

5.1 Approaches to systematic stabilization

The essential tool we will use in this chapter is the Poincaré map introduced in Section 2.2. We noted that local exponential stability of a periodic orbit $\mathcal{O}$ intersecting the Poincaré section $\mathcal{S}$ transversally can be expressed in terms of the spectral radius of the Jacobian of the Poincaré map evaluated at $x^* = \bar{\mathcal{O}} \cap \mathcal{S}$. Thus it is natural to address stability of periodic orbits in an optimization problem for which either the objective or the constraint includes

the spectral radius of the Jacobian of the Poincaré map. For example, several authors [42,65] propose optimization problems the form

$$\begin{aligned} \underset{\zeta}{\text{minimize}} \quad & \rho(A(\zeta)) \\ \text{subject to} \quad & g_E(\zeta) = 0 \\ & g_I(\zeta) \geq 0, \end{aligned} \tag{5.1}$$

where ζ combines parameters of the periodic orbit and the feedback controller, $A(\zeta)$ is the Jacobian of the Poincaré map of the closed-loop system, $\rho(A)$ is the spectral radius of A (or an approximation to it), and $g_E(\zeta)$ and $g_I(\zeta)$ are constraint functions which ensure that the solution is periodic and feasible for the physical system.

This formulation has the potential advantage of choosing simultaneously both the periodic orbit *and* a stabilizing feedback control law. Unfortunately, including the spectral radius generally makes the problem non-smooth and non-convex. Consider, for example, the problem of minimizing the spectral radius of $A(\xi) = \left[\begin{smallmatrix} 0 & 1-\xi \\ \xi & 1 \end{smallmatrix}\right]$ over the scalar ξ. The eigenvalues of $A(\xi)$ are $\frac{1}{2} \pm \sqrt{\frac{1}{4} - (\xi^2 - \xi)}$; the corresponding spectral radius is shown in Figure 5.1. For this example the spectral radius even fails to be Lipschitz.

Another drawback is that computing the Poincaré map is often computationally expensive. As there generally is no closed-form expression for the Poincaré map nor for its Jacobian, these are usually obtained by numerical integration of the closed-loop dynamics together with numerical differentiation. Thus a significant question will be how to cast the problem in a form that facilitates numerical solution of the resulting optimization problem.

Previous work in this area has highlighted several different approaches for finding stable periodic orbits of a hybrid system. Chevallereau et al. [42] formulated a gait design problem using HZD, which allows stability of a resulting periodic orbit to be analyzed using a restricted Poincaré map of dimension one less than the dimension of the zero dynamics man-

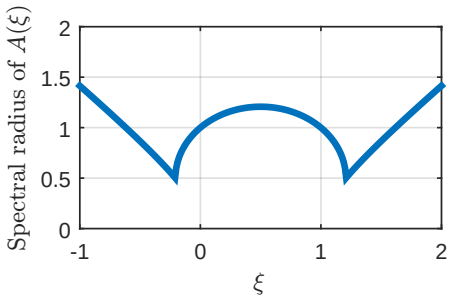


Figure 5.1: Spectral radius of $A(\xi)$ as a function of ξ.

ifold. As the Jacobian of the restricted Poincaré map can be computed much more efficiently than that of the full map, this method allowed for direct optimization of the spectral radius of the linearized Poincaré map during the gait design process for a rigid, five-link 3D biped model with restricted yaw. This method is especially attractive for systems with few degrees of underactuation, since underactuation increases the dimension of the zero dynamics manifold and hence of the restricted Poincaré map.

An alternative approach was proposed by Diehl et al. [65], which introduced a smoothed approximation to the spectral radius. Here, again, gait design was formulated as an optimization problem. The objective function included the smoothed spectral radius and a regularization term to penalize the use of excessive actuator torque. It was shown that the smoothed spectral radius can be evaluated efficiently by solving a relaxed Lyapunov equation. However, the computational burden of computing the linearized Poincaré map apparently remains. The method was demonstrated for the design of an open-loop stable walking gait for a planar two-link walking mechanism.

Here we address the stabilization problem by itself, assuming that the gait design process has already produced a suitable periodic orbit. We present a method to optimize the stability of the orbit without the need to recompute the Poincaré map at each iteration of the optimization algorithm. The key to this method is to restrict attention to parameterized families of feedback controllers satisfying a given set of hypotheses, which allows us to

approximate the Poincaré map by a truncated Taylor series.

Importantly, the hypotheses will ensure that for each choice of controller parameters, the same periodic orbit is a solution of the resulting closed-loop system. Thus, while the Poincaré map is, in general, a function of the controller parameters, its fixed point is not. The parameterized, linearized Poincaré map induces a parameterized discrete-time linear system for which stability of the origin is equivalent to a certain bilinear matrix inequality (BMI). A BMI optimization problem is then formulated to choose the controller parameters.

5.2 Orbital stability as an optimization problem

5.2.1 Problem formulation

Consider the system with impulse effects

$$\Sigma\colon \begin{cases} \dot{x} = f(x,\xi), & x^- \notin \mathcal{S} \\ x^+ = \Delta(x^-,\xi), & x^- \in \mathcal{S} \end{cases} \tag{5.2}$$

with flow f and reset map Δ parameterized by $\xi \in \Xi$, where Ξ is an open, connected subset of $\mathbb{R}^p$. We assume the state space $\mathcal{X}$ is an open, connected subset of $\mathbb{R}^n$, and that $f\colon \mathcal{X} \times \Xi \to \mathbb{R}^n$ and the impact map $\Delta\colon \mathcal{S} \times \Xi \to \mathcal{X}$ are $\mathcal{C}^2$. We also assume that $\mathcal{S}$ is nonempty and there exists a $\mathcal{C}^1$ function $\sigma\colon \mathcal{X} \to \mathbb{R}$ such that

$$\mathcal{S} = \{\, x \in \mathcal{X} \mid \sigma(x) = 0,\ \dot{\sigma}(x) < 0 \,\} \tag{5.3}$$

and 0 is a regular value of σ. Thus $\mathcal{S}$ is an embedded submanifold of $\mathcal{X}$ of codimension one.

We denote the unique solution of $\dot{x} = f(x,\xi)$ starting from $x(0) = x_0$ by $\varphi(t, x_0, \xi)$, for all $t \geq 0$ such that the solution exists. We also define the time-to-reset map $T\colon \mathcal{X} \times \Xi \to \mathbb{R}_{\geq 0}$

as the first time at which the solution $\varphi(t, x_0, \xi)$ intersects the switching manifold $\mathcal{S}$, i.e.,

$$T(x_0, \xi) := \inf \{\, t \geq 0 \mid \varphi(t, x_0, \xi) \in \mathcal{S} \,\}. \tag{5.4}$$

Our analysis assumes that there exists an orbit $\mathcal{O}$ transversal to $\mathcal{S}$ for the parameterized closed-loop hybrid model (5.2) which is independent of the controller parameters ξ. Formally, we require

Assumption 2 (Common periodic orbit)**.** *There exists a state $x_0^* \in \mathcal{X} \setminus \mathcal{S}$ such that*

(O1) *the solution of $\dot{x} = f(x, \xi)$ with $x(0) = x_0^*$ is independent of ξ for all $t \geq 0$;*

(O2) *the reset map evaluated at $x_f^* := \varphi(T(x_0^*, \xi), x_0^*, \xi)$ satisfies $\Delta(x_f^*, \xi) = x_0^*$ $\forall \xi \in \Xi$; and*

(O3) *the solution $\varphi(t, x_0^*, \xi), t \geq 0$ intersects $\mathcal{S}$ transversely.*

We denote the nominal solution of $\dot{x} = f(x, \xi)$ by $\varphi^*(t) := \varphi(t, x_0^*, \xi)$, the time to impact at x_0^* by $T^* := T(x_0^*, \xi)$, and the periodic orbit of the system with impulse effects (5.2) by $\mathcal{O} := \{\, \varphi^*(t) \mid 0 \leq t < T^* \,\}$.

5.2.2 Linearized Poincaré map for the parameterized system

Taking the switching manifold $\mathcal{S}$ as the Poincaré section, the Poincaré map $P \colon \mathcal{S} \times \Xi \to \mathcal{S}$ is defined by

$$P(x, \xi) := \varphi\left(T(\Delta(x, \xi), \xi), \Delta(x, \xi), \xi\right). \tag{5.5}$$

Transversality, together with differentiability of f and Δ, ensures that the Poincaré map is well defined and differentiable in a neighborhood of x_f^*. Furthermore, Assumption 2 implies that x_f^* is a fixed point of P for all $\xi \in \Xi$.

The Poincaré map allows us to study the behavior of solutions of the hybrid system by sampling the solution each time it intersects the impact manifold $\mathcal{S}$. It is convenient to

represent P in local coordinates. Let $\Lambda\colon U \to \mathbb{R}^{n-1}$ be a local coordinate system on an open subset $U \subset \mathcal{S}$ containing x_f^*. We can express the Poincaré map in the coordinates z defined by Λ as $\hat{P}(z,\xi) := \Lambda\left(P(\Lambda^{-1}(z),\xi)\right)$. Then for any choice of ξ, exponential stability of the periodic orbit $\mathcal{O}$ is equivalent [33] to exponential stability of $z_f^* = \Lambda(x_f^*)$ for the discrete-time system

$$z[k+1] = \hat{P}(z[k],\xi), \quad k = 0, 1, \ldots, \tag{5.6}$$

which is in turn equivalent to exponential stability of the origin for the linearized system

$$\delta z[k+1] = \frac{\partial \hat{P}}{\partial z}(z_f^*,\xi)\,\delta z[k], \quad k = 0, 1, \ldots, \tag{5.7}$$

where $\delta z[k] := z[k] - z_f^*$.

In order to exponentially stabilize the periodic orbit $\mathcal{O}$, we would like to choose ξ such that the spectral radius of $\frac{\partial \hat{P}}{\partial z}(z_f^*,\xi)$ is less than one. But as $\hat{P}$ depends nonlinearly on ξ, directly optimizing the spectral radius might be computationally expensive. As an alternative, we can expand the Taylor series approximation of $\hat{P}$ in (5.7). Since x_f^* is a fixed point of P for all $\xi \in \Xi$, it follows that

$$\frac{\partial \hat{P}}{\partial \xi}(z_f^*,\xi) = 0, \quad \forall \xi \in \Xi; \tag{5.8}$$

so we compute the sensitivity of $\frac{\partial \hat{P}}{\partial z}(z_f^*,\xi)$ with respect to ξ. For a sufficiently small perturbation $\Delta\xi$ about a nominal choice ξ^* of the parameters, (5.7) can by approximated by

$$\delta z[k+1] = \left(\frac{\partial \hat{P}}{\partial z}(z_f^*,\xi^*) + \sum_{i=1}^{p} \frac{\partial^2 \hat{P}}{\partial \xi_i \partial z}(z_f^*,\xi^*)\,\Delta\xi_i\right)\,\delta z[k], \tag{5.9}$$

where $\Delta\xi := (\Delta\xi_1, \cdots, \Delta\xi_p)^T := \xi - \xi^*$.

To simplify notation, we define[1]

$$A_0 := \frac{\partial \hat{P}}{\partial z}(z_f^*, \xi^*)$$
$$A_i := \frac{\partial^2 \hat{P}}{\partial \xi_i \partial z}(z_f^*, \xi^*), \quad i = 1, \ldots, p$$

and

$$A(\Delta\xi) := A_0 + \sum_{i=1}^{p} A_i \, \Delta\xi_i,$$

and we write (5.9) as

$$\delta z[k+1] = A(\Delta\xi)\, \delta z[k], \quad k = 0, 1, \ldots . \tag{5.10}$$

5.2.3 Optimization problem

Our objective is to choose values for $\Delta\xi$ that stabilize the orbit $\mathcal{O}$ of the closed-loop system (5.2). Provided $\Delta\xi$ is small, the Taylor series approximation (5.10) gives us a way to do this without having to recompute the Poincaré map at each iteration. We formulate the search for $\Delta\xi$ as an optimization problem which tries to minimize a weighted combination of the squared spectral radius of $A(\Delta\xi)$ and the 2-norm of $\Delta\xi$. Letting w be the scalar weight, we have the problem

$$\begin{aligned} \underset{\Delta\xi}{\text{minimize}} \quad & w\, \rho\left(A(\Delta\xi)\right)^2 + \|\Delta\xi\|_2^2 \\ \text{subject to} \quad & \rho\left(A(\Delta\xi)\right) < 1 \end{aligned} \tag{5.11}$$

Some comments on this optimization problem formulation are in order. First, we note that the optimization problem still makes sense when we set $w = 0$. This may be desirable when a large margin between the spectral radius and the unit circle is not expected to

[1]For comparison with [89, 90], note that $A(\Delta\xi)$ can also be written as $A_0 + \bar{A}(\Delta\xi \otimes I_{n\times n})$, where $\bar{A} = [A_1 \; A_2 \; \cdots \; A_p]$, or as $A_0 + B(I_{n\times n} \otimes \Delta\xi)$, where the columns of B are simply a permutation of the columns of $\bar{A}$. The symbol "$\otimes$" denotes the Kronecker product.

provide significant benefits.[2] Second, it is trivial to relax this problem (by omitting the constraint) when feasibility problems arise; the relaxed problem (with $w \neq 0$) may prove useful in providing a better nominal value of ξ about which to linearize the Poincaré map for subsequent constrained optimization. Finally, we note that this problem is not formally well-posed, since the infimum of the cost may not be attained on the open set defined by the constraints. Here and elsewhere, we will ignore such technicalities.

The framework we have introduced can be extended in various ways. For example, we could search for $\Delta\xi$ which simultaneously stabilizes a *collection* Ω of orbits. To do so, we would simply linearize the associated Poincaré maps and form an optimization problem with constraints $\rho(A^\omega(\Delta\xi)) < 1,\ \omega \in \Omega$ and cost $\|\Delta\xi\|_2^2 + \sum_{\omega\in\Omega} w_\omega\, \rho(A^\omega(\Delta\xi))^2$.

In the following sections we present two more extensions: First, we show how to incorporate a measure of robustness to uncertainty in the switching manifold into the optimization. We also extend the method for optimization with a restricted Poincaré map.

5.3 Disturbance rejection as an optimization problem

5.3.1 Problem formulation

A robot walking over rough terrain will experience impacts earlier or later than expected when walking on flat ground, which may cause it to deviate from its nominal gait. This motivates us to seek a method to minimize the effect of ground height disturbances on a walking gait.

To address this problem, we consider the system with impulse effects (5.2) with two modifications: First, we assume the impact map may be extended in a physically meaningful way to a $\mathcal{C}^2$ function $\Delta\colon \mathcal{X} \times \Xi \to \mathcal{X}$ on the whole state space (or at least on an open

[2]For example, Hobbelen and Wisse [72] demonstrated that the spectral radius is poorly correlated with the ability of a bipedal robot to reject disturbances.

neighborhood of $\mathcal{S}$ in $\mathcal{X}$). This is necessary in order to ascertain the effect of impacts occurring outside of $\mathcal{S}$. Second, we will assume that impacts occur when the state x intersects a generalized impact manifold

$$\mathcal{S}_d = \{\, x \in \mathcal{X} \mid \sigma(x) - d = 0,\ \dot{\sigma}(x) < 0 \,\}, \tag{5.12}$$

where d represents an input from the environment which is constant within each step. We will allow d to take values in a closed interval $\mathcal{D} := [-d_{\max}, d_{\max}] \subset \mathbb{R}$ for some $d_{\max} > 0$. For a model of bipedal walking, if σ represents the swing foot height, then an impact occurs in $\mathcal{S}_d$ when the robot steps up a height d. We may assume that the impact map does not depend explicitly on d; this assumption implies no loss of generality, since Δ depends on x and, at impacts, $d = \sigma(x)$. In what follows, we shall consider d as a *disturbance*.

Corresponding to the switching manifold $\mathcal{S}_d$ we define the *extended time-to-reset function* $T_e \colon \mathcal{X} \times \Xi \times \mathcal{D} \to \mathbb{R}_{\geq 0}$, given by

$$T_e(x_0, \xi, d) := \inf \{t > 0 \,|\, \varphi(t, x_0, \xi) \in \mathcal{S}_d\}\,, \tag{5.13}$$

and the *extended Poincaré map* $P_e \colon \mathcal{X} \times \Xi \times \mathcal{D} \to \mathcal{X}$, given by

$$P_e(x, \xi, d) := \varphi\left(T_e\left(\Delta(x, \xi), \xi, d\right), \Delta(x, \xi), \xi\right). \tag{5.14}$$

Formally, the extended Poincaré map defined here is not a Poincaré map at all, though its restriction to a codimension-1 submanifold is a hybrid version of the generalized Poincaré map defined in [74, Appendix D]; P_e maps a point $x \in \mathcal{X}$ to the next intersection of the trajectory with $\mathcal{S}_d$, *assuming an impact first occurs at the point* x.

Finally, we will augment the modified system with a performance output $h \colon \mathcal{X} \to \mathbb{R}^l$. The goal of the optimization problem will be to minimize the effect of the disturbance on

the performance output.

5.3.2 Discrete-time system for disturbance rejection

The map P_e defines the controlled discrete-time system

$$\begin{aligned} x[k+1] &= P_e\left(x[k], \xi, d[k]\right), \quad k = 0, 1, \dots \\ y[k] &= h(x[k]) \end{aligned} \tag{5.15}$$

in which $d[k] \in \mathcal{D}, k = 0, 1, \dots$ represents the disturbance input and $y[k]$ the performance output. Linearization of (5.15) around $(x_f^*, \xi, 0)$ results in

$$\begin{aligned} \delta x[k+1] &= \frac{\partial P_e}{\partial x}(x_f^*, \xi, 0)\, \delta x[k] + \frac{\partial P_e}{\partial d}(x_f^*, \xi, 0)\, d[k] \\ \delta y[k] &= \frac{\partial h}{\partial x}(x_f^*)\, \delta x[k]. \end{aligned} \tag{5.16}$$

Note that because P_e is defined on $\mathcal{X}$, we do not need to define a coordinate system on $\mathcal{S}$ before linearizing as we did in (5.7).

Once again, we may expand the Taylor series approximation of $\frac{\partial P_e}{\partial x}(x_f^*, \xi, 0)$ with the sensitivities of $\frac{\partial P_e}{\partial x}$ and $\frac{\partial P_e}{\partial d}$ to find

$$\begin{aligned} \delta x[k+1] = &\left(\frac{\partial P_e}{\partial x}(x_f^*, \xi^*, 0) + \sum_{i=1}^{p} \frac{\partial^2 P_e}{\partial \xi_i \partial x}(x_f^*, \xi^*, 0)\, \Delta\xi_i\right) \delta x[k] \\ &+ \left(\frac{\partial P_e}{\partial d}(x_f^*, \xi^*, 0) + \sum_{i=1}^{p} \frac{\partial^2 P_e}{\partial \xi_i \partial d}(x_f^*, \xi^*, 0)\, \Delta\xi_i\right) d[k] \\ \delta y[k] = &\frac{\partial h}{\partial x}(x_f^*)\, \delta x[k]. \end{aligned} \tag{5.17}$$

It is shown in [89, Section V] that the disturbance sensitivity matrix $\frac{\partial P_e}{\partial d}(x_f^*, \xi, 0)$ is indepen-

dent of ξ and hence, $\frac{\partial^2 P_e}{\partial \xi_i \partial d}(x_f^*, \xi^*, 0) = 0$. Consequently, (5.17) can be rewritten as follows

$$\begin{aligned} \delta x[k+1] &= A_e(\Delta\xi)\,\delta x[k] + B_e\, d[k] \\ \delta y[k] &= C_e\,\delta x[k], \end{aligned} \tag{5.18}$$

where

$$\begin{aligned} A_{0,e} &:= \frac{\partial P_e}{\partial x}(x_f^*, \xi^*, 0) \\ A_{i,e} &:= \frac{\partial^2 P_e}{\partial \xi_i \partial x}(x_f^*, \xi^*, 0), \quad i = 1, \ldots, p \\ B_e &:= \frac{\partial P_e}{\partial d}(x_f^*, \xi^*, 0) \\ C_e &:= \frac{\partial h}{\partial x}(x_f^*) \end{aligned}$$

and

$$A_e(\Delta\xi) = A_{0,e} + \sum_{i=1}^{p} A_{i,e}\Delta\xi_i.$$

5.3.3 Optimization problem

We now turn our attention to the disturbance rejection problem, which we describe first for the case of a bipedal robot. We consider the scenario (illustrated in Figure 5.2) where the robot starts on the nominal orbit at $x_f^* \in \mathcal{S}_0$, encounters a step of height $d[0] \in \mathcal{D}$ at the end of the first step, and finally takes one more step, this time not encountering any disturbance. We will try to minimize the effect of the disturbance in the first step on the value of the performance output at the end of the second step.

Remark. Our approach is quite similar to that introduced by Hobbelen and Wisse [72]. Specifically, our performance output is nothing more than a *gait indicator*. Both methods

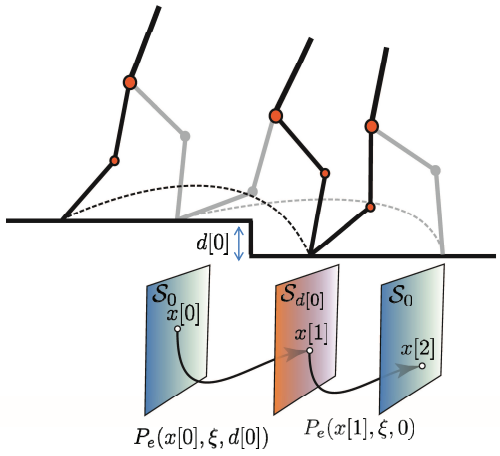


Figure 5.2: Illustration of the disturbance rejection problem for bipedal walking. The robot starts on the nominal orbit at $x[0] = x_f^* \in \mathcal{S}_0$. At the end of the first step, it encounters a step of height $d[0] \in \mathcal{D}$. It takes one more step, this time not encountering any disturbance.

quantify the effect of the disturbance on the performance output by linearizing the Poincaré map with a disturbance input. The primary difference is that the gait sensitivity norm defined in [72] computes the $\mathcal{H}_2$ norm of the transfer function from the disturbance to the output, while we focus on the single-event disturbance shown in Figure 5.2. One advantage of the gait sensitivity norm relative to our approach is that it considers the effect of the disturbance on all subsequent steps, which is not captured in our performance metric. On the other hand, we present a method to optimize the controller with respect to the performance metric. The optimization can be written as an nonlinear program, as is done in this section, or as a BMI optimization, as is done in Section 5.5.

For a general system, we assume that $d[0] \in \mathcal{D}$ is an *unknown* disturbance and $d[1] = 0$. The change in the performance output at time $k = 2$ caused by the disturbance $d[0]$ is given by $\delta y[2] = C_e\, A_e(\Delta\xi)\, B_e\, d[0]$. Minimizing the output over all possible disturbances yields

$$\min_{\Delta\xi} \max_{d[0]\in\mathcal{D}} \|\delta y[2]\|_2 = \min_{\Delta\xi} d_{\max}\, \|C_e\, A_e(\Delta\xi)\, B_e\|_2. \tag{5.19}$$

Combining this new objective with the original problem (5.11) yields the robust stability optimization problem

$$\begin{aligned} \underset{\Delta\xi}{\text{minimize}} \quad & w_1 \, \rho\left(A(\Delta\xi)\right)^2 + w_2 \, d_{\max}^2 \, \|C_e \, A_e(\Delta\xi) \, B_e\|_2^2 + \|\Delta\xi\|_2^2 \\ \text{subject to} \quad & \rho\left(A(\Delta\xi)\right) < 1 \end{aligned} \tag{5.20}$$

where w_1 and w_2 are positive weighting factors affecting the tradeoff between convergence rate, disturbance rejection capability, and perturbation size.

5.4 Systematic controller design using a restricted Poincaré map

We now specialize the presented optimization procedure for controller design using a restricted Poincaré map. The restricted Poincaré map takes advantage of time-scale separation and invariance induced by the underlying controller to isolate the dynamics on a submanifold of the state space. Using the restricted Poincaré map in conjunction with the optimization-based stabilization method offers several potential benefits. First, by focusing on the restriction dynamics, we can design stabilizing virtual constraints *independent of the controller used to stabilize the zero dynamics manifold*. This means that the BMI optimization can be used specifically to shape the zero dynamics. Of course, this simplifies subsequent tuning of the controller for the full model, since the stabilizing effect of the zero dynamics is independent of the feedback gains. Perhaps more importantly, by optimizing the virtual constraints directly, the optimization produces *more insightful results* in the sense that a control designer may be able to better interpret the optimized results. Finally, using the restricted Poincaré map reduces the size of the resulting optimization problem. This can result in much faster optimizations, and may make optimization feasible for systems of greater dimension than otherwise possible using available software packages.

The restricted Poincaré map originates in the context of hybrid zero dynamics, where

a feedback controller creates an invariant submanifold $\mathcal{Z}$ in the state space. It has been shown [33, 34] that under appropriate conditions, stability of an orbit of the closed-loop system can be determined by studying the stability of the same orbit for the system restricted to the invariant manifold. By introducing a *parameterized family of invariant manifolds*, we can systematically select one invariant manifold in order to stabilize an orbit. In order to meaningfully perform the sensitivity analysis required for the systematic selection of a submanifold, we must carefully consider the local smoothness of the parametrization. This is the objective of Sections 5.4.1 and 5.4.2, which essentially parallel the development in [34]. Indeed, our first result (Theorem 3) is partly a restatement of [34, Corollary 11] for a parameterized class of systems. The key novelty in the development here is the introduction of a smoothly parameterized output function (see (5.23)) and a smoothly parameterized family of coordinate charts (see (5.34)). Section 5.4.3 then presents expressions for the Jacobian of the restricted Poincaré map.

Notation. In the following, a map which has ξ as an explicit argument will be denoted with a subscript Ξ. Fixing a particular value of ξ defines a restricted map (see, for example, (5.23)), which will be denoted with a subscript ξ. Since these maps differ only in their domains, we will usually define explicitly only one of the maps (for example, the map $T_{I,\xi}\colon \mathcal{X} \times \mathcal{K} \to \bar{\mathbb{R}}_+$ in (5.29)), with the understanding that, by convention, any reference to the associated map (e.g., the extended map $T_{I,\Xi}(x, \kappa, \xi) := T_{I,\xi}(x, \kappa)$) is defined in relation to the first.[3]

[3]Formally, to define $T_{I,\Xi}$ as an extension of $T_{I,\xi}$ from $\mathcal{X} \times \mathcal{K}$ to $\mathcal{X} \times \mathcal{K} \times \Xi$, we must define the *family of maps* $\{T_{I,\xi}\}_{\xi \in \Xi}$.

5.4.1 Parameterized restriction dynamics

Consider the control system

$$\Sigma\colon \begin{cases} \dot{x} = f(x) + g(x)u, & x^- \notin \mathcal{S} \\ x^+ = \Delta(x^-), & x^- \in \mathcal{S} \end{cases} \tag{5.21}$$

with state $x \in \mathcal{X}$ and input $u \in \mathcal{U}$, where $\mathcal{X}$ and $\mathcal{U}$ are open, connected subsets of $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively. We assume $f\colon \mathcal{X} \to \mathbb{R}^n$, $g\colon \mathcal{X} \to \mathbb{R}^{n\times m}$, and $\Delta\colon \mathcal{S} \to \mathcal{X}$ are sufficiently smooth. We also assume that $\mathcal{S}$ is nonempty and there exists a sufficiently smooth function $\sigma\colon \mathcal{X} \to \mathbb{R}$ such that $\mathcal{S} = \{\, x \in \mathcal{X} \mid \sigma(x) = 0,\ \dot{\sigma} < 0 \,\}$ and 0 is a regular value of σ. Suppose $(\varphi^*(t), u^*(t)), t \geq 0$ is a periodic solution of Σ with minimal period $T^* > 0$ such that the closure of $\mathcal{O} := \{\, \varphi^*(t) \mid t \geq t_0 \,\}$ intersects $\mathcal{S}$ transversely at a single point $x_f^* := \bar{\mathcal{O}} \cap \mathcal{S}$.

In order to stabilize the orbit $\mathcal{O}$, we will define a parameterized family of submanifolds of the state space and a corresponding family of feedback laws which renders the submanifolds hybrid invariant. In general, achieving hybrid invariance requires event-based control. Thus we introduce a set of parameters (distinct from those which index the family of submanifolds and feedback laws) into the control system Σ for the purpose of achieving impact invariance. Denote these parameters by κ and suppose they take values in $\mathcal{K} \subset \mathbb{R}^r$. At each impact, the value of κ will be updated according to an input v determined by a rule v_Ξ which will be described shortly. Adjoining κ to Σ gives the hybrid extension

$$\Sigma_e\colon \begin{cases} (\dot{x}, \dot{\kappa}) = (f(x) + g(x)u, 0), & (x^-, \kappa^-) \notin \mathcal{S} \times \mathcal{K} \\ (x^+, \kappa^+) = \Big(\Delta(x^-), v\Big), & (x^-, \kappa^-) \in \mathcal{S} \times \mathcal{K}. \end{cases} \tag{5.22}$$

For convenience we define $x_e := (x, \kappa)$, $f_e(x_e) := (f(x), 0)$, and $g_e(x_e) := (g(x), 0)$.

Let Ξ be an open, connected subset of $\mathbb{R}^p$, and let $h_\Xi\colon \mathcal{X} \times \mathcal{K} \times \Xi \to \mathbb{R}^m$ be a sufficiently smooth function which vanishes on $\mathcal{O}_\Xi := \mathcal{O} \times \{\kappa^*\} \times \Xi$ for some $\kappa^* \in \mathcal{K}$. Suppose, further,

that for each $\xi \in \Xi$, the output function

$$h_\xi(x, \kappa) := h_\Xi(x, \kappa, \xi) \tag{5.23}$$

for Σ_e has uniform vector relative degree k in $\mathcal{X} \times \mathcal{K}$. Define

$$\mathcal{Z}_\Xi := \{\, (x, \kappa, \xi) \in \mathcal{X} \times \mathcal{K} \times \Xi \mid h_\xi(x, \kappa) = L_{f_e} h_\xi(x, \kappa) = \cdots = L_{f_e}^{k-1} h_\xi(x, \kappa) = 0 \,\}, \tag{5.24}$$

and, for each $\xi \in \Xi$,

$$\mathcal{Z}_\xi := \{\, (x, \kappa) \in \mathcal{X} \times \mathcal{K} \mid h_\xi(x, \kappa) = L_{f_e} h_\xi(x, \kappa) = \cdots = L_{f_e}^{k-1} h_\xi(x, \kappa) = 0 \,\}. \tag{5.25}$$

As h_Ξ vanishes on the (lifted) orbit, it follows that for each $\xi \in \Xi$, $\mathcal{Z}_\xi$ is not empty. Thus, for each $\xi \in \Xi$, $\mathcal{Z}_\xi$ is a codimension-(km) embedded submanifold of $\mathcal{X} \times \mathcal{K}$. Furthermore, if the distribution $\operatorname{span}\{g_1(x), \ldots, g_m(x)\}$ is involutive, where g_i is the ith column of g, then the unique input $u = u_\xi^*(x, \kappa)$ which renders $\mathcal{Z}_\xi$ invariant under the forward dynamics of Σ_e is given by

$$u_\xi^*(x, \kappa) = -\left(L_g L_f^{k-1} h_\xi(x, \kappa)\right)^{-1} L_f^k h_\xi(x, \kappa). \tag{5.26}$$

A feedback law $\Gamma_\xi \colon \mathcal{X} \times \mathcal{K} \to \mathcal{U}$ which coincides with u_ξ^* in $\mathcal{Z}_\xi$ (i.e., for which $\Gamma_\xi(x, \kappa) = u_\xi^*(x, \kappa)$ for all $(x, \kappa) \in \mathcal{Z}_\xi$) will be said to be *compatible with* $\mathcal{Z}_\xi$. Note that $u^*(t) = u_\xi^*(\varphi(t), \kappa^*)$.

To achieve impact invariance, we will update the parameters κ in a step-to-step manner

[34, 42]. We let $v_\Xi \colon \mathcal{X} \times \Xi \to \mathcal{K}$ be a sufficiently smooth function such that

$$\begin{aligned} h_\Xi(\Delta(x), v_\Xi(x,\xi), \xi) &= 0 \\ L_{f_e} h_\Xi(\Delta(x), v_\Xi(x,\xi), \xi) &= 0 \\ &\vdots \\ L_{f_e}^{k-1} h_\Xi(\Delta(x), v_\Xi(x,\xi), \xi) &= 0 \end{aligned} \tag{5.27}$$

for every $(x, \kappa, \xi) \in (\mathcal{S} \times \mathcal{K} \times \Xi) \cap \mathcal{Z}_\Xi$. If v_Ξ satisfies (5.27) on the larger set $(\mathcal{S} \times \mathcal{K} \times \Xi)$, then v_Ξ is said to be *deadbeat*. Equation (5.27) represents km constraints; thus, in general, we need $r \geq km$ independent parameters κ to ensure impact invariance.

Now consider the parameterized extension Σ_e, together with a parameterized output function h_ξ having uniform vector relative degree k, a feedback law Γ_ξ compatible with the manifold $\mathcal{Z}_\xi$, and a parameter update law v_Ξ which satisfies (5.27). Together these define a parameterized family of closed-loop systems

$$\bar{\Sigma}_\xi \colon \begin{cases} (\dot{x}, \dot{\kappa}) = (f(x) + g(x)\,\Gamma_\xi(x,\kappa), 0)\,, & (x^-, \kappa^-) \notin \mathcal{S} \times \mathcal{K} \\ (x^+, \kappa^+) = \big(\Delta(x^-), v_\xi(x^-)\big)\,, & (x^-, \kappa^-) \in \mathcal{S} \times \mathcal{K}, \end{cases} \tag{5.28}$$

where, according to our convention, $v_\xi(x) := v_\Xi(x, \xi)$.

Denote the unique solution of $f_\xi(x,\kappa) := (f(x) + g(x)\,\Gamma_\xi(x,\kappa), 0)$ starting from (x,κ) by $\varphi_\xi(t, x, \kappa)$, the time to impact

$$T_{I,\xi}(x,\kappa) := \inf\{\, t \geq 0 \mid \varphi_\xi(t,x,\kappa) \in \mathcal{S} \times \mathcal{K}\,\}, \tag{5.29}$$

and the point of impact

$$\varphi_{T_{I,\xi}}(x,\kappa) := \varphi_\xi(T_{I,\xi}(x,\kappa), x, \kappa). \tag{5.30}$$

Note that $\varphi_\xi(t, \Delta(x_f^*), \kappa^*) = (\varphi^*(t), \kappa^*),\ 0 \leq t < T^*$; thus $\varphi_\xi(t, \Delta(x_f^*), \kappa^*)$ intersects $\mathcal{S} \times \mathcal{K}$

transversely at the single point $(x_f^*, \kappa^*) = \varphi_{T_{I,\xi}}(\Delta(x_f^*), \kappa^*)$.

5.4.2 Parameterized Poincaré map

We can now define the Poincaré map $P_\xi \colon \mathcal{S} \times \mathcal{K} \to \mathcal{S} \times \mathcal{K}$ for $\bar{\Sigma}_\xi$ as

$$P_\xi(x, \kappa) := \varphi_{T_{I,\xi}} \circ \Delta_\xi(x, \kappa), \tag{5.31}$$

where

$$\Delta_\xi(x, \kappa) := (\Delta(x), v_\xi(x)) \,. \tag{5.32}$$

By construction, $\mathcal{Z}_\xi$ is hybrid invariant for the closed-loop system $\bar{\Sigma}_\xi$, so we can also define the restriction dynamics $\bar{\Sigma}_\xi|_{\mathcal{Z}_\xi}$ and the restricted Poincaré map $P_\xi|_{\mathcal{Z}_\xi} \colon \mathcal{Z}_\xi \cap (\mathcal{S} \times \mathcal{K}) \to \mathcal{Z}_\xi \cap (\mathcal{S} \times \mathcal{K})$. However, the domain of this restricted Poincaré map has dimension $n - km - 1 + r \geq n - 1$ due to the presence of the parameters κ. If the output function h_Ξ is chosen such that the intersection $\mathcal{Z}_\xi \cap (\mathcal{S} \times \mathcal{K})$ is a product $(\tilde{\mathcal{Z}}_\xi \cap \mathcal{S}) \times \mathcal{K}$ for some $\tilde{\mathcal{Z}}_\xi \subset \mathcal{X}$, then it is more convenient to define the restricted Poincaré map as the function $\rho_\xi \colon \tilde{\mathcal{Z}}_\xi \to \tilde{\mathcal{Z}}_\xi$ satisfying

$$P_\xi(x, \kappa) = (\rho_\xi(x),\, v_\xi(x)), \quad \forall (x, \kappa) \in (\tilde{\mathcal{Z}}_\xi \cap \mathcal{S}) \times \mathcal{K}. \tag{5.33}$$

The domain of ρ_ξ has dimension $n - km - 1$.

Let ξ^* be a nominal choice of the parameter vector ξ. Suppose there is a sufficiently smooth function $\Lambda_\Xi \colon U_\Xi \to \mathbb{R}^{n-km-1} \times \mathbb{R}^r \times \mathbb{R}^{km}$, defined on a neighborhood $U_\Xi \subset \mathcal{S} \times \mathcal{K} \times \Xi$ of (x_f^*, κ^*, ξ^*) and partitioned as $\Lambda_\Xi = (\Lambda_{\Xi,1},\, \Lambda_{\Xi,2},\, \Lambda_{\Xi,3})$, such that for all $\xi \in \Xi$, the function $\Lambda_\xi \colon U_\xi \to \mathbb{R}^{n-km-1} \times \mathbb{R}^r \times \mathbb{R}^{km}$ defined by

$$\begin{aligned} \Lambda_\xi(x, \kappa) &:= (\Lambda_{\Xi,1}(x, \kappa, \xi),\, \Lambda_{\Xi,2}(x, \kappa, \xi),\, \Lambda_{\Xi,3}(x, \kappa, \xi)) \\ &=: (\Lambda_{\xi,1}(x, \kappa),\, \Lambda_{\xi,2}(x, \kappa),\, \Lambda_{\xi,3}(x, \kappa)) \end{aligned} \tag{5.34}$$

is a diffeomorphism on the set U_ξ satisfying[4] $U_\xi \times \{\xi\} = U_\Xi \cap (\mathcal{S} \times \mathcal{K} \times \{\xi\})$, and Λ_ξ satisfies the following:

- $\Lambda_{\xi,1}$ and $\Lambda_{\xi,3}$ are independent of κ;
- $\Lambda_{\xi,2}(x, \kappa) = \kappa$; and
- $\Lambda_{\xi,3}(x, \kappa) = 0 \iff x \in \tilde{\mathcal{Z}}_\xi$.

We will call such a pair (U_Ξ, Λ_Ξ) a *parameterized coordinate chart*. Despite the terminology, (U_Ξ, Λ_Ξ) is not, itself, a coordinate chart, as the dimension of U_Ξ exceeds that of the codomain of Λ_Ξ by p; but for each $\xi \in \Xi$, the pair (U_ξ, Λ_ξ) *is* a coordinate chart on the projected ξ-slice U_ξ. Furthermore, (U_ξ, Λ_ξ) represents $U_\xi \cap (\tilde{\mathcal{Z}}_\xi \times \mathcal{K})$ as a slice of U_ξ.

The representation $\hat{P}_\xi$ of P_ξ in the coordinates (ζ, κ, η) defined by Λ_ξ is given by $\hat{P}_\xi := \Lambda_\xi \circ \varphi_{T_{I,\xi}} \circ \Delta_\xi \circ \Lambda_\xi^{-1}$. Similarly, the representation $\hat{\rho}_\xi$ of ρ_ξ in the coordinates ζ defined by Λ_ξ is given by[5] $\hat{\rho}_\xi(\zeta) := \Lambda_{\xi,1} \circ \varphi_{T_{I,\xi}} \circ \Delta_\xi \circ \Lambda_\xi^{-1}(\zeta, \kappa^*, 0)$. The following theorem describes the structure of the linearized Poincaré map for the closed-loop system $\bar{\Sigma}_\xi$.

Theorem 3 (Structure of the linearized Poincaré map)**.** *Consider the parameterized family of closed-loop systems $\{\bar{\Sigma}_\xi\}_{\xi \in \Xi}$, where $\bar{\Sigma}_\xi$ is formed from Σ_e in closed-loop with the feedback law $\Gamma_\xi(x, \kappa) := \Gamma_\Xi(x, \kappa, \xi)$ and update law $v_\xi(x) := v_\Xi(x, \xi)$ for sufficiently smooth functions $\Gamma_\Xi \colon \mathcal{X} \times \mathcal{K} \times \Xi \to \mathcal{U}$ and $v_\Xi \colon \mathcal{X} \times \Xi$. Denote the corresponding family of Poincaré maps by $\{P_\xi\}_{\xi \in \Xi}$. Suppose the following additional hypotheses are satisfied:*

(H1) *There is a non-trivial periodic orbit $\mathcal{O}$ of Σ whose closure intersects $\mathcal{S}$ transversely at a single point x_f^*.*

(H2) *There is a sufficiently smooth function $h_\Xi \colon \mathcal{X} \times \mathcal{K} \times \Xi \to \mathbb{R}^m$ which vanishes on $\mathcal{O} \times \{\kappa^*\} \times \Xi$ for some $\kappa^* \in \mathcal{K}$.*

[4] That is, U_ξ is the projection onto $\mathcal{S} \times \mathcal{K}$ of the ξ-slice of U_Ξ.

[5] Note that we could use any other value $\kappa \in \mathcal{K}$ in the definition of $\hat{\rho}_\xi$, since this value is ignored by Δ_ξ.

(H3) *For each* $\xi \in \Xi$*, the function* h_ξ *defined by* (5.23) *has uniform vector relative degree* k *in* $\mathcal{X} \times \mathcal{K}$*.*

(H4) *For each* $\xi \in \Xi$*, there exists a submanifold* $\tilde{\mathcal{Z}}_\xi \subset \mathcal{X}$ *such that* $\mathcal{Z}_\xi \cap (\mathcal{S} \times \mathcal{K}) = (\tilde{\mathcal{Z}}_\xi \cap \mathcal{S}) \times \mathcal{K}$*; furthermore,* $\tilde{\mathcal{Z}}_\xi \cap \mathcal{S}$ *is a codimension-*(km) *embedded submanifold of* $\mathcal{S}$*.*

(H5) *For each* $\xi \in \Xi$*, the feedback law* Γ_ξ *is compatible with* $\mathcal{Z}_\xi$*.*

(H6) *The parameter update law* v_Ξ *satisfies* (5.27) *for every* $(x, \kappa, \xi) \in (\mathcal{S} \times \mathcal{K} \times \Xi) \cap \mathcal{Z}_\Xi$*.*

(H7) *There exists a parameterized coordinate chart* (U_Ξ, Λ_Ξ) *defined on a neighborhood* $U_\Xi \subset \mathcal{S} \times \mathcal{K} \times \Xi$ *of* (x_f^*, κ^*, ξ^*) *for some nominal parameter vector* ξ^**.*

Then, for any $\xi \in \Xi$*, there exists a neighborhood* $V_\xi \subset \mathcal{X} \times \mathcal{K}$ *of* $(\Delta(x_f^*), \kappa^*)$ *and a coordinate transformation* $\Psi_\xi \colon V_\xi \to \mathbb{R}^{n-km} \times \mathbb{R}^r \times \mathbb{R}^{km}$ *such that when* P_ξ *is represented in the coordinates* $z_\xi = (\zeta, \kappa, \eta)$ *defined by* Λ_ξ*, its Jacobian about the fixed point* $(\zeta^*, \kappa^*, \eta^*) = \Lambda_\xi(x_f^*, \kappa^*)$ *is*

$$\frac{\partial \hat{P}_\xi}{\partial z_\xi}(\zeta^*, \kappa^*, \eta^*) = \begin{bmatrix} \frac{\partial \hat{\rho}_\xi}{\partial \zeta}(\zeta^*) & 0 & \star \\ \star & 0 & \star \\ 0 & 0 & S_{\phi_{T_{I,\xi}}}(\bar{\zeta}^*, \kappa^*, \eta^*) S_{\Delta_\xi}(\zeta^*, \kappa^*, \eta^*) \end{bmatrix}, \tag{5.35}$$

where $S_{\phi_{T_{I,\xi}}}(\bar{\zeta}^*, \kappa^*, \eta^*) := \mathcal{D}_3(\Lambda_{\xi,2} \circ \varphi_{T_{I,\xi}} \circ \Psi_\xi^{-1}(\bar{\zeta}^*, \kappa^*, \eta^*))$ *is the sensitivity of the transverse dynamics of the trajectory evaluated at* $(\bar{\zeta}^*, \kappa^*, \eta^*) = \Psi_\xi(\Delta(x_f^*), \kappa^*)$*, and* $S_{\Delta_\xi}(\zeta^*, \kappa^*, \eta^*) := \mathcal{D}_3(\Psi_{\xi,3} \circ \Delta_\xi \circ \Lambda_{e,\xi}^{-1}(\zeta^*, \kappa^*, \eta^*))$ *is the sensitivity of the transverse dynamics of the impact map.*

Furthermore, we can write the Jacobian of the parameterized restricted Poincaré map $\frac{\partial \hat{\rho}_\xi}{\partial \zeta}(\zeta^*)$ *as a differentiable function of the parameter vector* ξ*.*

The proof is sketched. For each $\xi \in \Xi$, existence of the coordinate chart (V_ξ, Ψ_ξ) and the form of $\frac{\partial \hat{P}_\xi}{\partial z_\xi}$ given by (5.35) essentially follow from [34, Corollary 11]. Specifically, hypotheses (H1)–(H3) imply that, for each $\xi \in \Xi$, the set $\mathcal{Z}_\xi$ defined by (5.25) is a (non-empty) codimension-(km) embedded submanifold of $\mathcal{X} \times \mathcal{K}$ [91, Lemma 5.1.1]. Hypothesis (H5)

ensures that Γ renders $\mathcal{Z}_\xi$ forward invariant, while (H6) ensures impact invariance. As the orbit $\mathcal{O}$ is contained in $\mathcal{Z}_\xi$ (by (H2)) and as Λ_ξ and Ψ_ξ both represent (open subsets of) $\mathcal{Z}_\xi$ as slices, it follows that $\eta^* = 0$ and $\hat{P}_\xi(\zeta_i, \kappa_i, \eta^*) = (\zeta_f, \kappa_f, \eta^*)$ for all $(\zeta_i, \kappa_i) \in \mathbb{R}^{n-km-1} \times \mathbb{R}^r$, whence the two block zeros in the bottom row of $\frac{\partial \hat{P}_\xi}{\partial z_\xi}$. Furthermore, P_ξ is independent of κ (by (5.31)), whence the block zeros in the middle column of $\frac{\partial \hat{P}_\xi}{\partial z_\xi}$. Differentiability of the Jacobian $\frac{\partial \hat{\rho}_\xi}{\partial \zeta}(\zeta^*)$ is a consequence of the smoothness of the local parametrization by ξ.

The structure of the Jacobian of $\hat{P}_e$ implies that its spectrum is simply the union of the spectra of the diagonal blocks $\frac{\partial \hat{\rho}_\xi}{\partial \zeta}(\zeta^*)$, 0, and $S_{\phi_{T_{I,\xi}}}(\bar{\zeta}^*, \kappa^*, \eta^*) S_{\Delta_\xi}(\zeta^*, \kappa^*, \eta^*)$. As shown in [34], the spectra of $S_{\phi_{T_{I,\xi}}}(\bar{\zeta}^*, \kappa^*, \eta^*)$ and $S_{\Delta_\xi}(\zeta^*, \kappa^*, \eta^*)$ can be made arbitrarily small by adjusting, respectively, the feedback gains and the parameter update law. (In fact, if the parameter update law is chosen to be deadbeat, then $S_{\Delta_\xi}(\zeta^*, \kappa^*, \eta^*) = 0$.) Thus, to stabilize the periodic orbit, we need only choose ξ so that the spectral radius of the Jacobian of the restricted Poincaré map is less than one. The next subsection derives an expression for this Jacobian.

5.4.3 Computing the linearized restricted Poincaré map

The linearized Poincaré map depends on ξ through the feedback law Γ_ξ, the update law v_ξ, and the coordinate transformation Λ_ξ. We have carefully constructed each of these elements so that their composition is a differentiable function of ξ. Suppose the dependence of Ψ_ξ on ξ is also nice; that is, suppose there is a sufficiently smooth function Ψ_Ξ such that $\Psi_\Xi(x, \kappa, \xi) = \Psi_\xi(x, \kappa)$. Then we can make the dependence on ξ more explicit by defining

$$\hat{P}_\Xi(\zeta, \kappa, \eta, \xi) := (\Lambda_\Xi \circ \varphi_{T_{I,\Xi}} \circ \Delta_\Xi \circ \bar{\Lambda}_\Xi^{-1})(\zeta, \kappa, \eta, \xi), \tag{5.36}$$

where the definitions of $\varphi_{T_{I,\Xi}}$ and Δ_Ξ follow from (5.29) and (5.32) according to our convention, and $\bar{\Lambda}_\Xi(x, \kappa, \xi) := (\Lambda_\Xi(x, \kappa, \xi),\ \xi)$.

We can further expand $\hat{P}_\Xi$ as

$$\hat{P}_\Xi(\zeta,\kappa,\eta,\xi) = (\Lambda_\Xi \circ \varphi_{T_I,\Xi} \circ \bar{\Psi}_\Xi^{-1}) \circ (\bar{\Psi}_\Xi \circ \Delta_\Xi \circ \bar{\Lambda}_\Xi^{-1})(\zeta,\kappa,\eta,\xi), \tag{5.37}$$

where $\bar{\Psi}_\Xi(x,\kappa,\xi) := (\Psi_\Xi(x,\kappa,\xi),\, \xi)$. The arguments proving the preceding theorem imply that the Jacobians S_{Φ_Ξ} and S_{Δ_Ξ} of $(\Lambda_\Xi \circ \varphi_{T_I,\Xi} \circ \bar{\Psi}_\Xi^{-1})$ and $(\bar{\Psi}_\Xi \circ \Delta_\Xi \circ \bar{\Lambda}_\Xi^{-1})$, when partitioned compatible with $z_\Xi := (\zeta,\kappa,\eta,\xi)$, have the form

$$\frac{\hat{P}_\Xi}{\partial z_\Xi}(\zeta^*,\kappa^*,\eta^*,\xi^*) = \begin{bmatrix} S_{\Phi_{\Xi,11}} & S_{\Phi_{\Xi,12}} & S_{\Phi_{\Xi,13}} & S_{\Phi_{\Xi,14}} \\ 0 & I & 0 & 0 \\ 0 & 0 & S_{\Phi_{\Xi,33}} & 0 \end{bmatrix} \begin{bmatrix} S_{\Delta_{\Xi,11}} & 0 & S_{\Delta_{\Xi,13}} & 0 \\ S_{\Delta_{\Xi,21}} & 0 & S_{\Delta_{\Xi,23}} & S_{\Delta_{\Xi,24}} \\ 0 & 0 & S_{\Delta_{\Xi,33}} & 0 \\ 0 & 0 & 0 & I \end{bmatrix}. \tag{5.38}$$

Standard results imply that the Jacobian of $(\Lambda_\Xi \circ \varphi_{T_I,\Xi} \circ \bar{\Psi}_\Xi^{-1})$ can be expressed as

$$(\Lambda_\Xi \circ \varphi_{T_I,\Xi} \circ \bar{\Psi}_\Xi^{-1}) = \left(I - \left(\frac{\partial\hat{\sigma}}{\partial z_\xi}\hat{f}_\xi\right)^{-1} \hat{f}_\xi \frac{\partial\hat{\sigma}_\xi}{\partial z_\xi}\right) \frac{\partial\hat{\varphi}_T}{\partial z_\Xi}, \tag{5.39}$$

where $\hat{f}_\xi := \mathcal{D}\Lambda_\xi f_\xi \circ \Lambda_\xi^{-1}$ and $\hat{\sigma} := \sigma \circ \Lambda_\xi$ are the representations of f_ξ and σ in local coordinates, $\frac{\partial\hat{\varphi}_T}{\partial z_\Xi}$ is an extended monodromy matrix (i.e., the monodromy matrix with an additional block column corresponding to the parameters ξ), $z_\xi := (\zeta,\kappa,\eta)$, and $z_\Xi := (\zeta,\kappa,\eta,\Xi)$. The Jacobian of the restricted Poincaré map can be expressed as the upper left block of the product (5.38):

$$\frac{\partial\hat{\rho}}{\partial\zeta}(\zeta^*,\xi) = S_{\Phi_{\Xi,11}} S_{\Delta_{\Xi,11}} + S_{\Phi_{\Xi,12}} S_{\Delta_{\Xi,21}}. \tag{5.40}$$

5.5 Formulation via bilinear matrix inequalities

We now present alternative formulations of the optimization problems (5.11) and (5.20) with linear cost functions and *linear* and *bilinear matrix inequality* (LMI/BMI) constraints.

For a given choice of $\Delta\xi$, stability of the origin for (5.10) is equivalent to a certain matrix inequality, as shown by the following theorem. The proof is an application of the Schur complement lemma, which asserts that the pair of matrix inequalities $Q(x) > 0,\ R(x) - S(x)^T Q(x)^{-1} S(x) > 0$ is equivalent to the single matrix inequality

$$\begin{bmatrix} Q(x) & S(x) \\ S(x)^T & R(x) \end{bmatrix} > 0, \tag{5.41}$$

with $Q(x)$ and $R(x)$ symmetric. See [92] for a proof of an alternative version of Schur's lemma.[6]

Theorem 4 (BMI for Stabilization of the Origin)**.** *For fixed $\Delta\xi \in \mathbb{R}^p$, the origin $\delta z = 0$ is exponentially stable for* (5.10) *if there exist a scalar $\mu > 0$ and a symmetric matrix $W \in \mathbb{R}^{n\times n}$ such that*

$$\begin{bmatrix} W & A(\Delta\xi)W \\ WA(\Delta\xi)^T & (1-\mu)W \end{bmatrix} > 0. \tag{5.42}$$

Proof. Schur's lemma and (5.42) imply that $W > 0$ and $WA(\Delta\xi)^T W^{-1} A(\Delta\xi)W - (1-\mu)W < 0$, or, multiplying on the left and right by W^{-1},

$$A(\Delta\xi)^T W^{-1} A(\Delta\xi) - (1-\mu)W^{-1} < 0. \tag{5.43}$$

But (5.43) implies that the Lyapunov function $V[k] := \delta z[k]^T W^{-1} \delta z[k]$ satisfies $V[k+1] <$

[6]The two statements of the lemma differ by a similarity transformation which exchanges the roles of Q and R, and those of S and S^T. It is also assumed in [92] that $Q(x)$, $R(x)$, and $S(x)$ are affine in x; this assumption implies that (5.41) is a *linear* matrix inequality, but is otherwise not required. In our case (5.41) will be *bilinear*.

$(1-\mu)V[k]$. As $W^{-1} > 0$, this proves $\|\delta z[k]\|_2^2 < (1-\mu)^k \kappa \|\delta z[0]\|_2^2$ where κ is ratio of the largest to the smallest eigenvalue (i.e., the condition number) of W^{-1}. $\square$

The preceding theorem gives sufficient conditions for exponential stability, provided $\Delta\xi$ is not so large that the Taylor series approximation (5.10) fails. Noting that, by Schur's lemma, $\|\Delta\xi\|_2^2 < \gamma$ is equivalent to $\left[\begin{smallmatrix} I & \Delta\xi \\ \Delta\xi^T & \gamma \end{smallmatrix}\right] > 0$, we have the following optimization problem

$$\begin{aligned} \underset{W,\Delta\xi,\mu,\gamma}{\text{minimize}} \quad & -w\,\mu + \gamma \\ \text{subject to} \quad & \begin{bmatrix} W & A(\Delta\xi)W \\ WA(\Delta\xi)^T & (1-\mu)W \end{bmatrix} > 0 \\ & \begin{bmatrix} I_{p\times p} & \Delta\xi \\ \Delta\xi^T & \gamma \end{bmatrix} > 0 \\ & \mu > 0. \end{aligned} \tag{5.44}$$

We now demonstrate the equivalence between the BMI formulation (5.44) and the NLP formulation (5.11). Suppose $(W, \Delta\xi, \mu, \gamma)$ is feasible for the BMI problem. Then by Theorem 4, $\Delta\xi$ is feasible for the NLP. Conversely, if $\Delta\xi$ is feasible for the NLP, then for any $0 < \mu < 1 - \rho(A(\Delta\xi))$, the Lyapunov equation $A(\Delta\xi)^T X A(\Delta\xi) - (1-\mu)X = 0$ has a unique positive definite solution X. Letting $W = X^{-1}$ and $\gamma > \|\Delta\xi\|_2^2$, it follows that $(W, \Delta\xi, \mu, \gamma)$ is feasible for the BMI problem.

Now let $\Delta\xi$ be such that $A(\Delta\xi)$ is Hurwitz. The BMI optimization will drive $\mu \nearrow 1 - \rho(A(\Delta\xi))^2$ and $\gamma \searrow \|\Delta\xi\|_2^2$, so the limiting cost of $(W, \Delta\xi, \mu, \gamma)$ for the BMI problem will differ from the cost of $\Delta\xi$ for the NLP only by a constant $(-w)$. This proves that the NLP and the BMI are equivalent.

Similarly, the robust stability optimization problem (5.20) may be expressed in terms of

LMIs and BMIs as

$$
\begin{aligned}
\underset{W,\Delta\xi,\mu,\gamma,\eta}{\text{minimize}} \quad & -w_1\,\mu + w_2\eta + \gamma \\
\text{subject to} \quad & \begin{bmatrix} W & A(\Delta\xi)W \\ WA(\Delta\xi)^T & (1-\mu)W \end{bmatrix} > 0 \\
& \begin{bmatrix} I_{l\times l} & C_e A_e(\Delta\xi)B_e \\ B_e^T A_e(\Delta\xi)^T C_e^T & \eta/d_{\max}^2 \end{bmatrix} > 0 \\
& \begin{bmatrix} I_{p\times p} & \Delta\xi \\ \Delta\xi^T & \gamma \end{bmatrix} > 0 \\
& \mu > 0.
\end{aligned}
\tag{5.45}
$$

5.6 Solver performance comparison

We have shown how the search for stabilizing virtual constraints may be formulated as an optimization problem with bilinear matrix inequality constraints. Other well-known examples of control problems which can be expressed in terms of BMIs include robust, low-order, and decentralized feedback controller synthesis problems [92, 93]. The potential utility of BMIs for solving a variety of challenging control problems has motivated research into efficient BMI solvers. Unfortunately, BMIs are, in general, nonlinear, non-smooth, and non-convex; furthermore the BMI feasibility problem (that is, the problem of finding a feasible solution to a particular BMI) is known to be NP-hard [94]. Nevertheless, BMI optimization problems still possess special structure which can be exploited by specialized solvers. For example, Hassibi, How, and Boyd [95] proposed an iterative method for solving BMI problems. The proposed method uses a linearized approximation of the BMI at each step, resulting in a semi-definite program which, being convex, can be solved efficiently. The optimization problems which appear in subsequent chapters were solved using PENBMI [96]

together with YALMIP [97], which provides a MATLAB interface to general purpose solvers.

One potential benefit of the BMI formulation is the linear cost and the structured constraints. In principle, these could make the optimization problem easier to solve. However, the structure comes at a cost: the NLP (5.11) has only p free variables, while (5.44) has $n(n+1)/2 + p + 2$. Both formulations are non-smooth and non-convex.

To understand how these formulations compare, we examined the performance of two solvers (PENBMI and MATLAB's `fmincon`) for the problems (5.44) and (5.11), respectively, for different choices of n and p. The problem data $A_i, i = 0, 1, \ldots, p$ were randomly generated with elements independently drawn from a zero mean Gaussian with standard deviation[7] $0.4\sqrt{n}$. For each (n, p) pair we generated twenty problems and solved them with both solvers. We then computed the ratio of the median computation times, and the difference in the median achieved cost. Figure 5.3 illustrates the results.

For these test problems, the BMI solver is much slower than the NLP solver except when n is very small. Furthermore, the optimal cost to which the BMI solver converged is in almost every case greater than the optimal cost found by the NLP solver. This may indicate that the solver is not fully exploiting the structure of the problem (5.44). That structure could potentially be exploited to great advantage by iterative schemes which, for example, alternately solve a Lyapunov equation (with $\Delta\xi$ fixed) and an LMI optimization (with W fixed). On the other hand, that either solver successfully minimizes the cost function in a reasonable amount of time for much larger problems (in terms of n) than those solved in [42,65] suggests that Taylor series approximation of the Poincaré map is a very worthwhile tool for controller design.

There are obvious limitations to this parametric study. First, there are numerous options and tolerances which could be adjusted to alter the performance of each solver; we have not made that effort here, except to significantly relax the tolerance for the BMI solver. Second,

[7]Empirically, this scaling of the standard deviation normalizes the expected spectral radius of A_i to 0.4.

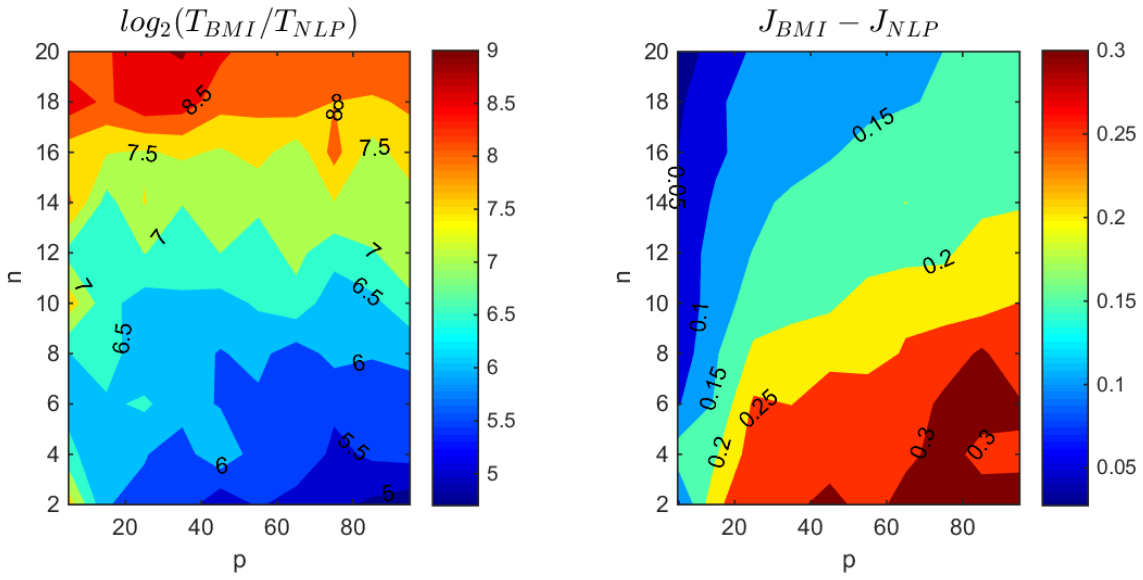


Figure 5.3: BMI versus NLP solver performance comparison. The plots compare computation time T and optimal cost J for BMI and NLP problem formulations as a function of the problem size.

we have used default (or zero) initial conditions; yet there may be much better methods for initializing each solver. For example, the BMI might be initialized by finding the smallest perturbation (if it exists) in a single element of $\Delta\xi$ which stabilizes the system, then solving a Lyapunov equation to find an initial matrix W. A third limitation is that the randomly generated data almost certainly fail to represent the structure of most practical problems. Fourth, we have not compared the performance for the robust stability problem (5.45). Fifth, state-of-the-art solvers for general NLPs are well established, while efficient solvers for BMI problems are still relatively immature. It is expected that continued research will yield significant improvements for BMI optimization.

CHAPTER 6

Stable 3D Walking with Hybrid Zero Dynamics and Systematic Optimization

The experiments described in Chapter 4 represent an advance in underactuated dynamic walking. The dynamic nature of the walking gaits achieved is perhaps comparable only to PETMAN and its successor ATLAS, both of which feature additional sensing and actuation in the ankles. Furthermore, these first successful walking experiments represent the initial experimental validation of the utility of virtual constraints for 3D bipedal robot walking.

In this chapter we build on the experimental success of Chapter 4 and the theoretical developments of Chapter 5. Our purpose is two-fold: First, we demonstrate the use of the optimization-based method for systematic design of stabilizing virtual constraints for MARLO. Second, we contribute to the validation of the methods of hybrid zero dynamics for 3D bipedal walking by bringing the gait and controller design closer to the experimental implementation. Both of these objectives support our aim to establish systematic gait and controller design methods for bipedal robots.

In contrast to the gait presented in Chapter 4, which was hand-tuned based on earlier planar walking, the gait design presented in this chapter is based on the 3D robot model. The gait design is realized by setting up an optimization to find a feasible periodic orbit and

a set of virtual constraints for the 3D model. In this stage we incorporate the intuitively-developed controlled coordinates (described in Section 4.5) for lateral stabilization. We analyze the stability of the resulting orbit using the method of Poincaré, and optimize the controlled coordinates using the method developed in Section 5.2. The resulting controller is tested both in simulation and experiments with MARLO.

6.1 Virtual constraints for 3D walking

As we discovered in Chapter 4, the choice of controlled variables in the lateral plane is critical for successful walking. Intuition and experience both suggest that stability is unlikely to be achieved when virtual constraints ignore lateral motion of the robot. This motivated the SIMBICON-like controlled variable defined by (4.18). While this lateral swing hip virtual constraint effectively stabilized the walking gait, tuning the parameters of the the output was a delicate task.

From (4.18), the controlled variable used previously is equivalent to

$$h_{0,sw}(q) = \left(1-(1-s)^3\right) q_{3,sw} - 3(1-s)^2 s b_{sw} + \left(3s^2-2s^3\right)\left(a(1+c_p)q_{yT} + \psi_{sw,d0} - c_p q_{3,st}\right), \quad (6.1)$$

where s is the normalized mechanical phase variable (see (4.4)). For gait design and subsequent controller optimization, it is more convenient to choose a controlled variable that is linear in q. To that end we consider (6.1) with $b_{sw} = \psi_{sw,d0} = 0$, $c_p = 0.85$, and $s = 1$.

In the sagittal plane, we use the same controlled variables as in Chapter 4 (see (4.2)).

The complete set of controlled variables is

$$h_0(q) = \begin{bmatrix} q_{\mathrm{grLA},R} \\ q_{\mathrm{grLA},L} \\ q_{\mathrm{grKA},R} \\ q_{\mathrm{grKA},L} \\ q_{\mathrm{HA},R} \\ h_{0,sw} \end{bmatrix} = \begin{bmatrix} \frac{q_{\mathrm{gr1R}}+q_{\mathrm{gr2R}}}{2} \\ \frac{q_{\mathrm{gr1L}}+q_{\mathrm{gr2L}}}{2} \\ q_{\mathrm{gr2R}} - q_{\mathrm{gr1R}} \\ q_{\mathrm{gr2L}} - q_{\mathrm{gr1L}} \\ q_{3R} \\ q_{3L} - (1+c_p)q_{yT} - c_p q_{3R} \end{bmatrix}. \tag{6.2}$$

Since h_0 is linear, we can also express it as $H^* q$ where $H^* \in \mathbb{R}^{m \times N}$ is full rank.

As before, the mechanical phase variable θ is taken to be the angle of the virtual stance leg relative to the ground (see (4.3)).

Specification of the virtual constraints defines the submanifold $\mathcal{Z}$ on which the hybrid zero dynamics evolves. This submanifold has dimension $2(N-m) = 6$, where $N = 9$ is the number of degrees of freedom of the model without springs in single support and $m = 6$ is the number of virtual constraints. We define coordinates on $\mathcal{Z}$ as

$$z = \begin{bmatrix} q_{\mathrm{zT}} \\ q_{\mathrm{yT}} \\ \theta \\ \dot{q}_{\mathrm{zT}} \\ \dot{q}_{\mathrm{yT}} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} q_{\mathrm{zT}} \\ q_{\mathrm{yT}} \\ \frac{3\pi}{2} - q_{\mathrm{xT}} - q_{\mathrm{grLA},st} \\ \dot{q}_{\mathrm{zT}} \\ \dot{q}_{\mathrm{yT}} \\ -\dot{q}_{\mathrm{xT}} - \dot{q}_{\mathrm{grLA},st} \end{bmatrix}, \tag{6.3}$$

where $q_{\mathrm{grLA},st} = q_{\mathrm{grLA},R}$ in right stance and $q_{\mathrm{grLA},st} = q_{\mathrm{grLA},L}$ in left stance.

Table 6.1: Constraints for forward walking gait design

Inequality constraints	Equality constraints
workspace limits	periodicity
torque limits	foot height at impact
bounds on $\Delta\theta$	hip constraints
foot clearance	$s^- = 1$
final swing foot velocity	
positive vertical impulse	
minimum horizontal COM velocity	
minimum forward COM displacement	

6.2 Gait design through optimization

As before, we let h_d be a vector of 5th order Bézier polynomials. Then the gait design process consists of finding a set of Bézier parameters α and an initial condition $z(0)$ which together yield a periodic orbit of the hybrid restriction dynamics. For this we set up a constrained nonlinear optimization. The constraints include upper and lower bounds on the optimization variables, as well as the inequality and equality constraints listed in Table 6.1. These constraints are used to ensure that the resulting periodic orbit will be physically meaningful. The "hip constraints" listed in the table represent equality constraints on several Bézier parameters relating to the desired hip angles; their purpose is to limit the aggressiveness of the desired hip motion. We do not constrain the mechanical phase variable θ to be monotonic along the periodic orbit; we simply verify afterward that this is the case for the optimized gait.

The cost function is chosen as the norm of the equality constraint violation. This choice is a good first step for finding feasible gaits, which is our goal here. It is also common to optimize for energetic efficiency [84] or robustness [73].

The initial condition for the optimizer is based on the experimental walking gait from

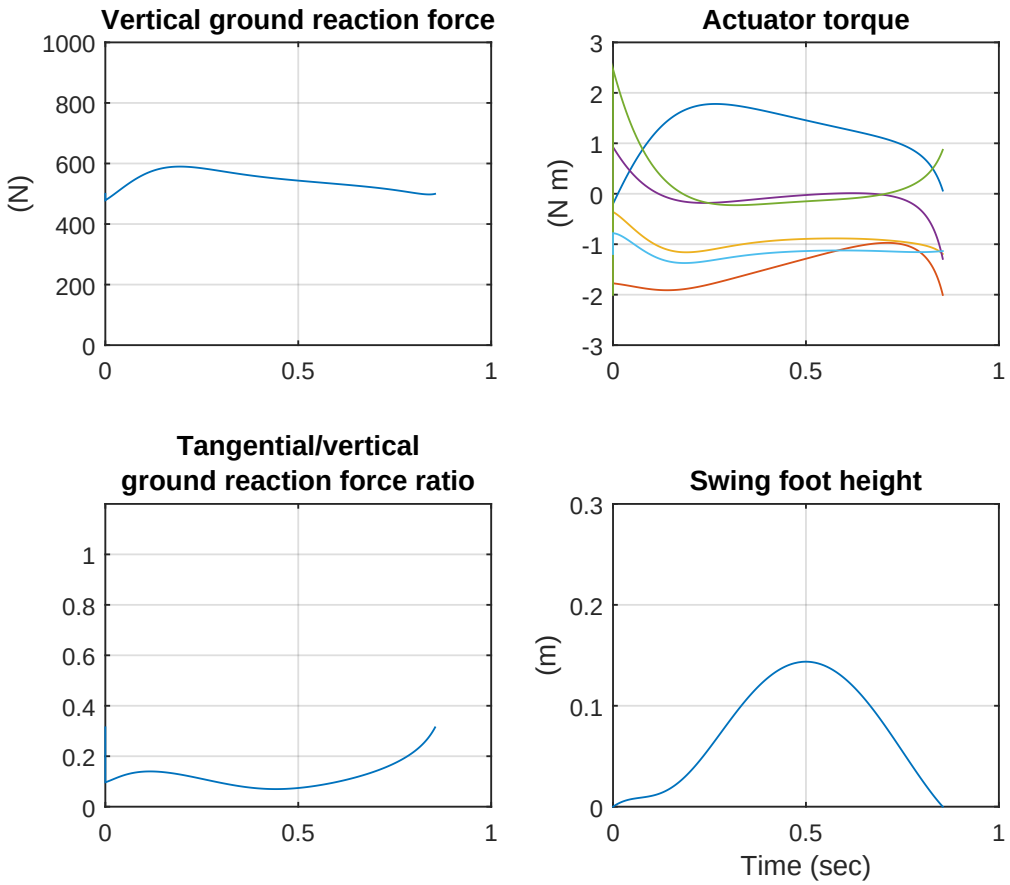


Figure 6.1: Feasibility of the walking gait.

Section 4.5.

Figure 6.1 illustrates the feasibility of the designed gait. Ground reaction forces remain safely within the friction cone, with the maximum ratio of tangential to vertical forces less than 0.4. All motor torques remain less than 3 Nm, which is the continuous torque limit for MARLO's amplifiers. The swing foot height is more than adequate for flat ground walking.

Figure 6.2 shows phase portraits for the designed gait in right stance. From these plots it is easy to see that the gait is periodic (modulo yaw). The robot yaws approximately 11 degrees over the course of one step. Because the model and gait are symmetric, the robot

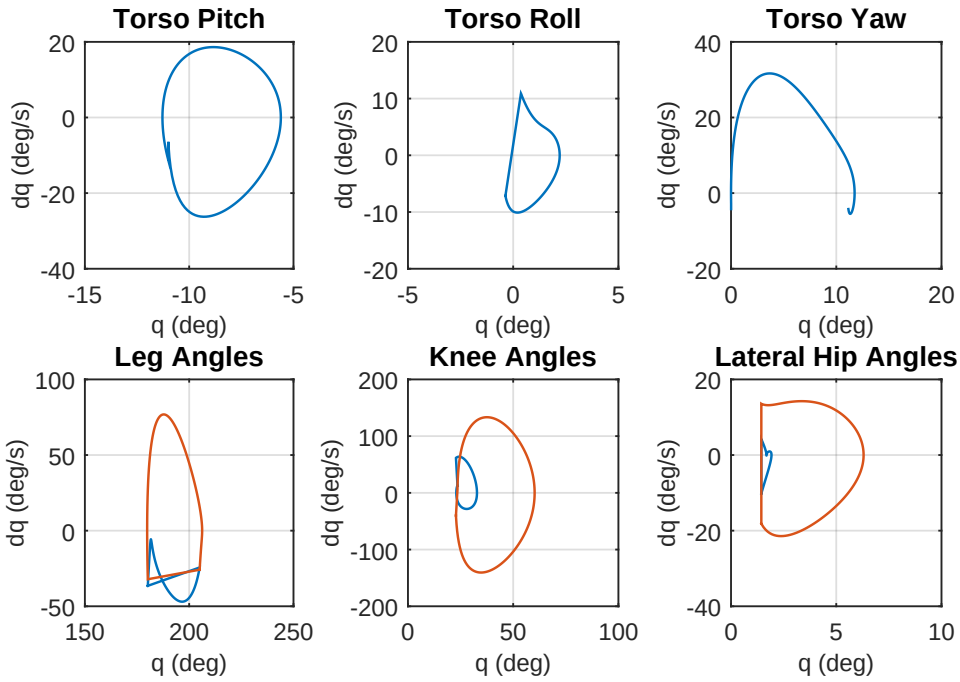


Figure 6.2: Phase portraits for the walking gait.

will yaw 11 degrees in the opposite direction during the following left stance phase. With the current foot model, which includes only viscous damping of yaw at the stance foot, achieving a significant reduction in the yaw motion would potentially require the robot to perform awkward motions, which is undesirable. In the experiments, where stiction at the foot is also present, the yaw motion will lead to the robot executing a slow turn over the course of multiple steps.

In our gait design we have used the model without springs. Figure 6.3 illustrates a comparison of two simulations: one using the rigid model, and one using the model with springs (assuming a spring stiffness of 7500 Nm/rad and a damping ratio of 0.5). The trajectories from the two models are in very good agreement, with the greatest deviation (1.3 degrees) occurring shortly after impact.

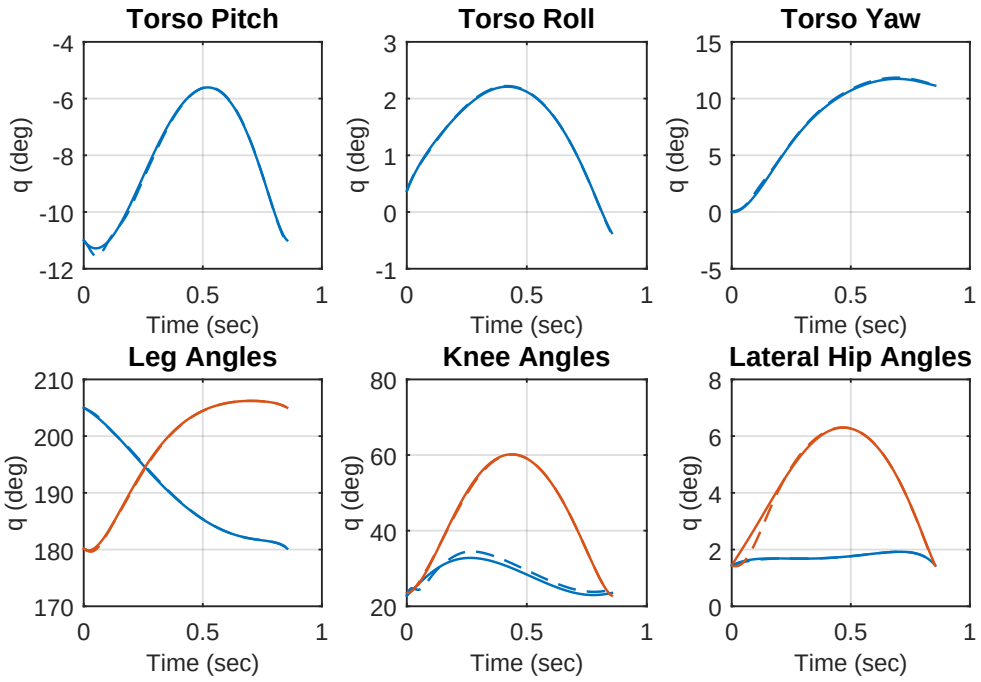


Figure 6.3: Comparison between the models with and without springs. Time trajectories for the model without springs are solid, while trajectories for the model with springs are dashed.

6.3 Closed-loop design and stability analysis

The gait design process yields a periodic orbit which lies in the zero dynamics manifold $\mathcal{Z}$. It is the task of the feedback controller to enforce the virtual constraints by rendering $\mathcal{Z}$ attractive and invariant under the forward dynamics of the closed-loop system. To create a hybrid zero dynamics, the controller must also ensure invariance of $\mathcal{Z}$ under the impact map. Here we present a feedback control law which achieves hybrid invariance, allowing us to study stability of the periodic orbit using the restricted Poincaré map. We also present and analyze several variations on the control law which will be used in experiments. These variations do not ensure invariance of $\mathcal{Z}$, but will nevertheless approximately enforce the virtual constraints and may do so more safely in the presence of inevitable model uncertainty.

6.3.1 Feedback linearization

Recall that the input-output linearizing feedback (2.15) renders $\mathcal{Z}$ forward invariant and attractive. Using this feedback law in simulation permits us to study the effects of different choices of virtual constraints on the stability of an orbit. Because exact (forward) invariance is achieved, when this feedback law is combined with the event-based updates described below, we can isolate the stabilizing effect of the choice of virtual constraints from the effect of PD control. However, as noted earlier feedback linearization is known to be sensitive to parametric uncertainty in the model. For this reason we will only use (2.15) for gait design and stability analysis.

6.3.2 PD + feedforward

Following [84], we make use of a modified version of (2.15) for experimental implementation. The modification consists of substituting regressed torques for u^* and a constant matrix T for the decoupling matrix $L_g L_f h$. The feedforward torque is determined from the simulation model by regressing the torques along the periodic orbit as 5th order Bézier polynomials in the normalized mechanical phase variable s. Thus the feedback law used is given by

$$u_{\text{exp}} = u^*(s, \alpha_\tau) - T^{-1}\left(\frac{K_P}{\varepsilon^2} y + \frac{K_D}{\varepsilon}\dot{y}\right) \tag{6.4}$$

where α_τ represents the Bézier coefficients of the feedforward torque.

6.3.3 Event-based update

For hybrid invariance, we employ event-based updates to the output function h, as described in Section 2.4. Formally, this creates a deadbeat hybrid extension [34] of the system, with the parameters α of the desired evolution and the initial value θ_i^+ of the mechanical phase variable being updated at the beginning of each step and remaining constant until

the next impact. After each impact, we compute the value $h_0^+ = h_0(q^+)$ and derivative $\dot{h}_0^+ = \frac{\partial h_0}{\partial q}(q^+)\dot{q}^+$ of the controlled coordinates from the post-impact state $(q^+, \dot{q}^+)$. We then update the phase normalization

$$s(\theta) = \frac{\theta - \theta_i^+}{\theta^- - \theta_i^+}, \tag{6.5}$$

with the initial value $\theta_i^+ = \theta(q^+)$ of the mechanical phase variable in step i. The parameters α of the desired evolutions are then updated so that

$$\begin{cases} h_d(\theta(q^+), \alpha) = h_0^+ \\ \frac{\partial h_d}{\partial \theta}(\theta(q^+), \alpha)\frac{\partial \theta}{\partial q}\dot{q}^+ = \dot{h}_0^+ \\ h_d(\theta_f, \alpha) = h_d(\theta_f, \alpha^*), \end{cases} \tag{6.6}$$

where α^* is the nominal parameter set. This update law ensures that the post-impact error and its derivative are zero.

6.3.4 Stability analysis

To evaluate the stability of the designed gait under various choices of feedback we compute the linearized Poincaré maps of the corresponding closed-loop systems. Jacobians are estimated by symmetric differences with a uniform step size of 10^{-4} radians. Feedback gains K_P, K_D, and ε were chosen based on walking experiments.

Feedback linearization with event-based update. Hybrid invariance, together with the deadbeat nature of the event-based update law, imply that the non-zero eigenvalues of the linearized Poincaré map for the closed-loop system will simply be the eigenvalues of the linearized restricted Poincaré map, which are $\{-1.84, -1, 0.75, -0.49, 0.43\}$. The eigenvalue -1 corresponds to yaw, and is expected as neither the robot dynamics nor the feedback controller depend on yaw [43, Prop. 4].

Feedback linearization without event-based update. For this feedback law the dominant eigenvalues are $\{-1.64, -1, 0.75, -0.46, 0.35\}$. This feedback law will be used in conjunction with BMI optimization to determine how the virtual constraints should be modified to achieve exponential stability of the orbit.

PD + feedforward with event-based update. For experiments, we implement only the approximate feedback linearization using the regressed feedforward torques and constant decoupling matrix. When event-based updates are used in conjunction with this feedback law, the dominant eigenvalues become $\{-1.99, -1, 0.72, -0.54, 0.24\}$. One motivation for using event-based updates even when the continuous feedback law does not ensure forward invariance is to reduce the magnitude of discontinuities in the torque during experiments.

PD + feedforward without event-based update. Disabling the event-based updates leads to a simpler control law which nevertheless approximately enforces the virtual constraints. The dominant eigenvalues in this case are $\{-1.77, -1, 0.74, -0.54, 0.23\}$.

6.4 Optimization of controlled coordinates

In order to use the optimization-based methods of Chapter 5 to stabilize the orbit $\mathcal{O}$, we must first have a parameterized family of feedback controllers such that the closed-loop system satisfies Assumption 2. Specifically, $\mathcal{O}$ must be a solution of the closed-loop system independent of the choice of controller parameters. Since we already know that stability depends critically on the choice of controlled variables, it is natural to let the optimization refine our controlled variables. We could also parameterize the mechanical phase variable θ and the feedback gains K_P and K_D, but we do not pursue this here.

6.4.1 Parametrization of virtual constraints

From the gait design process we know the trajectory $\varphi(t), 0 \leq t < T$ of the system along a periodic orbit. As θ is monotonic along the trajectory, there is a function $q_d(\theta)$ such that$q_d(\theta(\varphi(t))) = \varphi(t), 0 \leq t < T$. It follows that, for any matrix H, the quantity $H\left(q - q_d(\theta(q))\right)$ vanishes along the orbit. Parameterizing H by ξ, we can define a *family* of virtual constraints[1]

$$y = h(q, \xi) = H(\xi)\left(q - q_d(\theta(q))\right). \tag{6.7}$$

Furthermore, the original constraints given by (6.2) are included in this family as

$$y = H^*\left(q - q_d(\theta(q))\right). \tag{6.8}$$

Since $h(q, \xi)$ vanishes along the orbit for any ξ, (6.7) induces a parameterized family of feedback controllers satisfying Assumption 2.

As a technical condition, we should require that the parameter set Ξ be chosen such that the regularity conditions of Lemma 1 (which ensure existence of the zero dynamics) are satisfied for all $\xi \in \Xi$. Since existence of the zero dynamics is equivalent to invertibility of the decoupling matrix

$$\frac{\partial h}{\partial q} D^{-1} B = H\left(I - \frac{\partial q_d}{\partial \theta}\frac{\partial \theta}{\partial q}\right) D^{-1} B$$

at every point along the orbit, we are apparently faced with the task of finding the set of $\xi \in \mathbb{R}^p$ such that no non-zero element of the row span of $H(\xi)$ lies in the left nullspace of $\left(I - \frac{\partial q_d}{\partial \theta}\frac{\partial \theta}{\partial q}\right) D^{-1} B$ at any point along the orbit. In practice, it is easier simply to check invertibility of the decoupling matrix for the optimized result.

For the experiments reported here, we consider a parametrization of H which allows each virtual constraint to depend on the torso roll angle. The motivation for this is that the torso

[1]Note that we have omitted the argument α since it will be fixed for a given orbit.

pitch is already coupled into the virtual constraints through the mechanical phase variable, but the roll angle is not. We hypothesize that by allowing the virtual constraints to vary with the roll angle, the robot will be able to compensate appropriately when its roll deviates from the nominal orbit. We therefore have

$$H(\xi) = \begin{bmatrix} 0 & \xi_1 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & \xi_2 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \xi_3 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & \xi_4 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & \xi_5 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \xi_6 - (1 + c_p) & 0 & 0 & 0 & -c_p & 0 & 0 & 1 \end{bmatrix}. \tag{6.9}$$

The nominal controlled coordinates in (6.2) are simply $H^*q = H(\xi^*)q$, where $\xi^* = 0$ is the nominal choice of parameters.

Because we have chosen not to make the virtual constraints yaw-dependent, the linearized Poincaré map of the closed-loop system will have an eigenvalue of -1 for all values of ξ. Thus, to proceed with an optimization "modulo yaw", we must eliminate the yaw coordinate, which we do by a simple projection. Specifically, we compute the Jacobian $A_0 = \frac{\partial P}{\partial x}(x_f^*, \xi^*)$ of the corresponding Poincaré map, and the sensitivities $A_i,\ i = 1, \ldots, 6$ of A_0 with respect to perturbations in ξ_i. We then remove the first row and column of each. The resulting matrices are assembled into the affine matrix function $A(\Delta\xi) = A_0 + A_1\Delta\xi_1 + \cdots + A_6\Delta\xi_6$ comprising the model data needed for the BMI optimization problem (5.44). Using PENBMI and YALMIP we solve this optimization problem with the cost weight $w = 10$.

6.4.2 Results

The optimal perturbation of ξ is found to be $\Delta\xi = (-0.26, 0.20, 0.30, -0.23, -0.06, 0.24)$, and the corresponding spectral radius of $A(\Delta\xi)$ is 0.28.

When the revised virtual constraints are used in each of the closed-loop systems described in Section 6.3.4, the eigenvalues of the linearized Poincaré map are: Feedback linearization with event-based update: $\{-1, -0.68, 0.68, -0.32, 0.08\}$; Feedback linearization without event-based update: $\{-1, 0.58, -0.42, -0.42, 0.32\}$; PD + feedforward with event-based update: $\{-1, -0.34, -0.34, 0.56, -0.39\}$; PD + feedforward without event-based update: $\{-1, -0.34, -0.34, 0.53, -0.04\}$. As before, the eigenvalue -1 corresponds to yaw.

We see that the revised virtual constraints stabilize the orbit for the closed-loop system with any of the control laws considered.

6.5 Experimental evaluation

6.5.1 Method

The controller design presented in this chapter was evaluated on MARLO. We compared the controller based on the nominal virtual constraints with the BMI-optimized constraints. Experiments were performed on flat ground in the laboratory, where MARLO can walk approximately 7-8 meters during a single experiment.[2] A snapshot from one walking experiment is shown in Figure 6.4. During these experiments, the real-time computer was on-board the robot. Power was supplied by an off-board battery bank carried on a mobile gantry. The gantry is designed to catch the robot when power is cut at the end of an experiment, or in the event of an early failure. It does not support the robot or provide any stabilization during the walking experiments.

In each experiment the controller executes a gait initiation sequence as follows:

1. **Posing.** With the robot suspended from the gantry, the robot is posed in a neutral configuration with both legs side-by-side. The robot is then lowered to the ground and

[2]The lab is about 12.5 meters long; however, equipment near the perimeter of the lab and the need to pose the robot in front of the mobile gantry limit the maximum walking distance to about 8 meters.

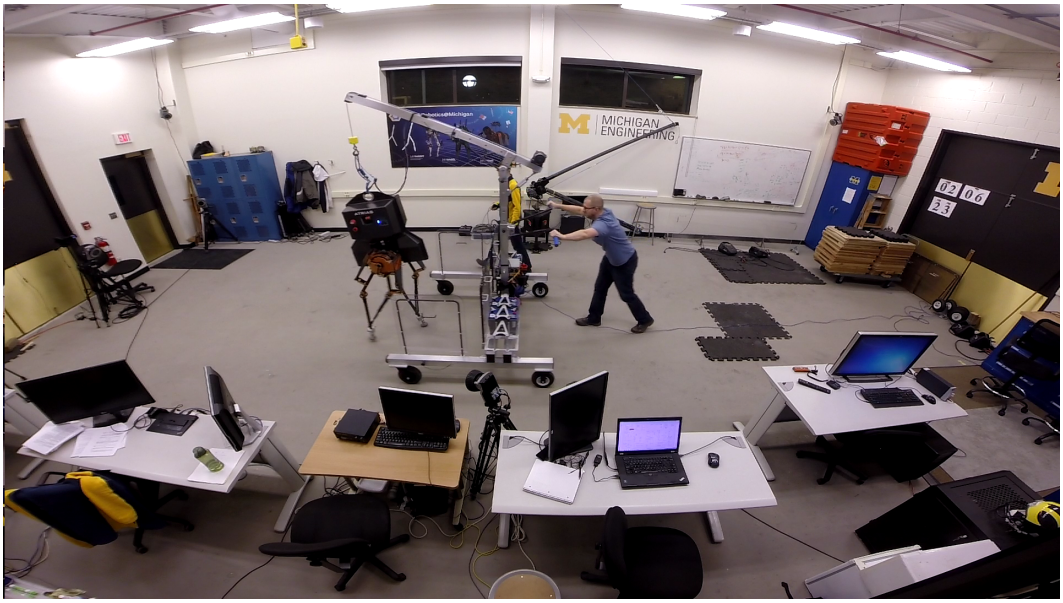


Figure 6.4: Laboratory setup for 3D walking experiments. The robot begins in Posing mode on the black mats.

begins supporting its own weight. Because the toroidal feet are not large enough to achieve static balance, an experimenter manually stabilizes the robot in the sagittal plane. He seeks to minimize the force required to hold the robot, ensuring that the COM is close to the center of the ground contact area. After the robot is nearly balanced, the controller waits for the experimenter to release the robot before entering the Injection phase. Release is detected by comparing the pitch rate to a pre-specified threshold (-3 degrees per second).

2. **Injection.** When the pitch rate crosses the threshold, the controller enters a short (300 ms) Injection phase. The goal during this phase is to initiate a lateral rocking motion away from the left leg so that the robot does not roll excessively on the first step. During this phase the controller rapidly extends the left knee 5 degrees from the posing configuration.

3. **Transition.** The controller then enters the Transition phase, in which it takes a

short first step. The goal of this phase is to accelerate the robot forward in order to approach the periodic orbit. The transition step employs hand-modified virtual constraints originally based on an optimization [46].

4. **Walking.** After a single transition step, the controller enters the Walking phase, where it remains for the duration of the experiment. Swing leg impact is detected using the knee angle spring deflection on the swing and stance legs. After each impact, the virtual constraint parameters are updated to zero the error y and its derivative $\dot{y}$. During the first five steps of the Walking phase, the torso is offset two degrees forward to help the robot gain speed.

During the Transition and Walking phases, control is based on virtual constraints of the form (4.2); only the Bézier parameters and the scaling of θ differ. During the Posing phase the controller regulates the controlled coordinates to a setpoint. In the injection phase, a time-based phase variable is used to ramp the swing knee angle.

Prior to the experiments reported here, we ran a series of experiments in which the virtual constraints were minimally adjusted to achieve walking. This is necessary due to persistent discrepancies between the model and the robot, most notably relating to the torso mass distribution and to friction in the harmonic drives. The swing knee angle virtual constraint, in particular, required the most tuning. We hypothesize that this is due to a combination of stiction in the harmonic drives and the rapid change in the desired knee angle. To address this we reduced the maximum desired knee angle. We also made the desired stance knee angle constant to reduce bouncing. Finally, the swing knee angle feedforward torque was adjusted by hand to improve tracking. These adjustments are shown in Figures 6.5 and 6.6. The -5 degree offset in leg angles shown in Figure 6.5 biases the torso backward. During the first two experiments reported below, the offset was -2 degrees.

In each experiment the robot was allowed to walk until: 1) the robot approached the

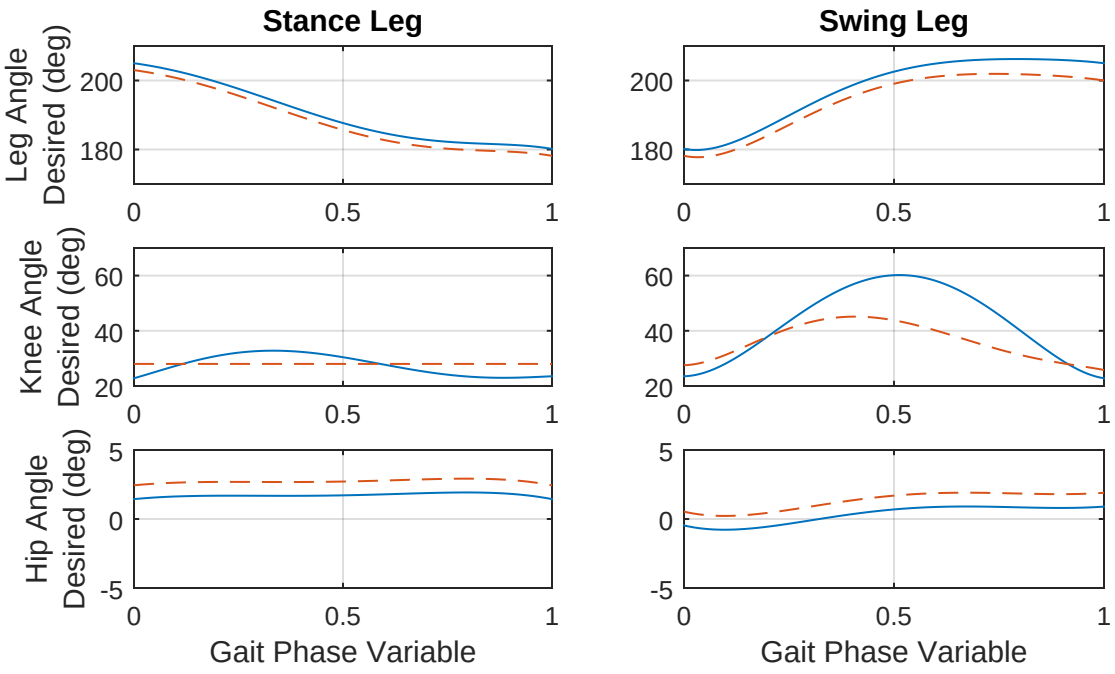


Figure 6.5: Comparison between nominal and modified virtual constraints. The nominal desired evolutions are solid, and the modified evolutions are dashed.

perimeter of the walking area; 2) the state of the robot left a (conservative) safe operating region; or 3) an experimenter cut motor power. The last of these occurred twice; in both cases the robot lost forward momentum and appeared to be on the verge of falling when the power was cut.

6.5.2 Results

We performed several walking experiments using both the nominal and the optimized controlled coordinates. Additionally, we tested each choice of controlled coordinates with and without event-based updates. A summary of these experiments is provided in Table 6.2.

Figure 6.7 shows the approximate path taken (i.e., the projection of the midpoint between the hips into the ground plane) during each of the experiments. Eight of the nine experiments using BMI-optimized outputs ended because the robot reached the limit of the walking area

Table 6.2: Summary of several walking experiments

ID	Date	Experiment number[a]	Controlled coordinates	Event-based update	Total steps	Reason ended
N1	Feb 6	22	nominal	enabled	14	power cut
B1		23	optimized	enabled	19	perimeter
B2	Feb 12	1	optimized	enabled	14	perimeter
N2		3	nominal	enabled	11	power cut
B3		4	optimized	enabled	4	safety
N3		6	nominal	enabled	10	safety[b]
B4		7	optimized	enabled	15	perimeter
N4		9	nominal	enabled	4	safety
B5		10	optimized	enabled	13	perimeter
N5		12	nominal	enabled	3	safety
B6	Feb 13	3	optimized	disabled	15	perimeter
B7		6	optimized	disabled	20	perimeter
N6		8	nominal	disabled	6	safety
N7		9	nominal	disabled	14	safety
B8		10	optimized	disabled	19	perimeter
B9		11	optimized	disabled	19	perimeter

[a] Experiments are numbered sequentially each day. The experiment numbers skipped in this table tested a different controller which is not reported here.
[b] Safety stop preceded by external disturbance from the safety cable.

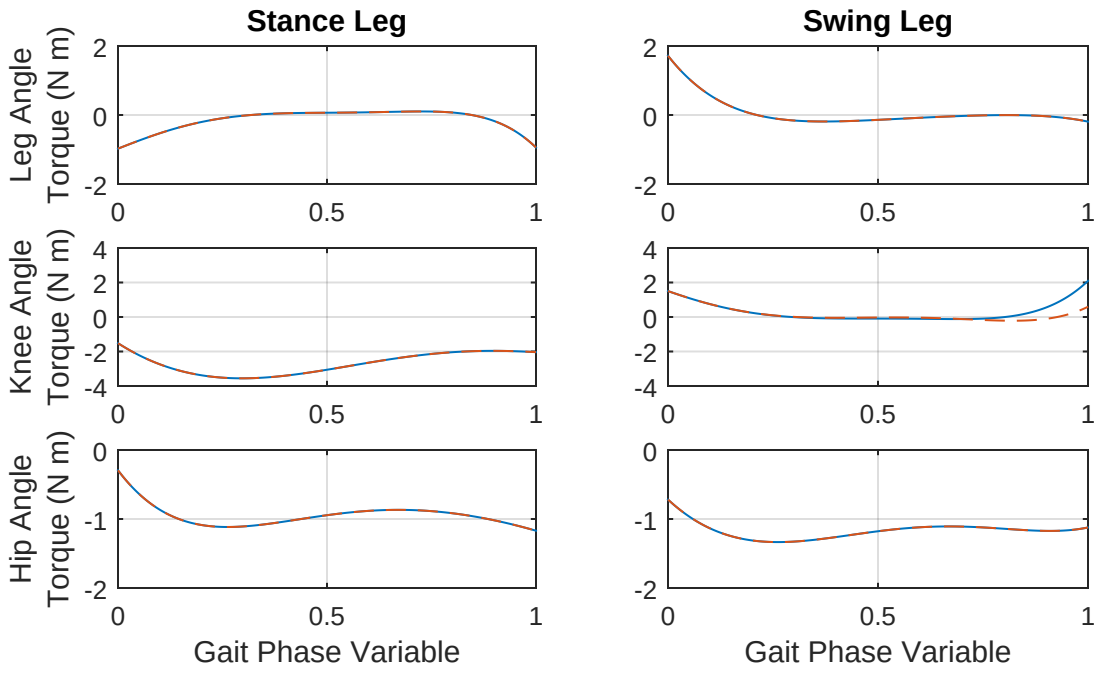


Figure 6.6: Comparison between nominal and modified feedforward torques. The nominal feedforward torques are solid, and the feedforward torques are dashed.

in the lab. As shown in the figure, the robot tended to turn gradually to the right; thus in most cases the robot reached the corner or the right wall instead of walking directly toward the far end of the lab. The robot tended to turn less when using the nominal outputs; however the effect may be partly due to the shorter distance walked: With the nominal outputs the robot did not reach the end of the lab. In most of the experiments with the nominal outputs, the controller software halted the robot because the state was outside the safe operating region. In two of these cases, the torso roll angle exceeded a threshold; in one case the norm of the output velocity $\dot{y}$ exceeded a threshold; and in two cases the hip angles were too narrow (meaning the feet might collide).

Figures 6.8 and 6.9 show the motion of the torso. Here the gradual turning is evident. The average yaw rate was around -9.8 degrees per second with the nominal outputs and -11.0 degrees per second with the optimized outputs. The torso pitch oscillates with each

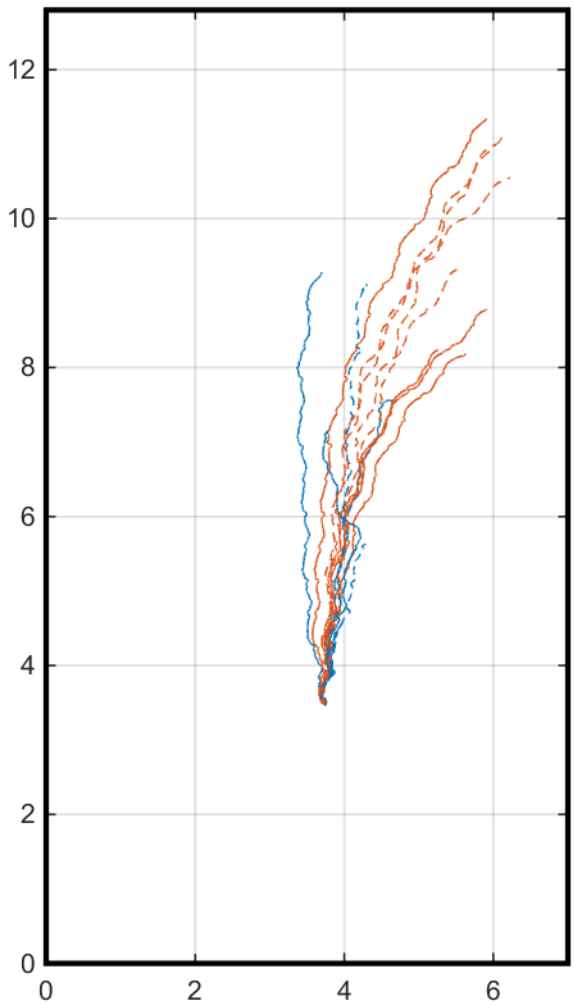


Figure 6.7: Approximate walking paths for 3D experiments. Experiments using the nominal outputs are traced in blue, and those using the optimized outputs are traced in red. Dashed lines indicate experiments in which event-based updates were disabled.

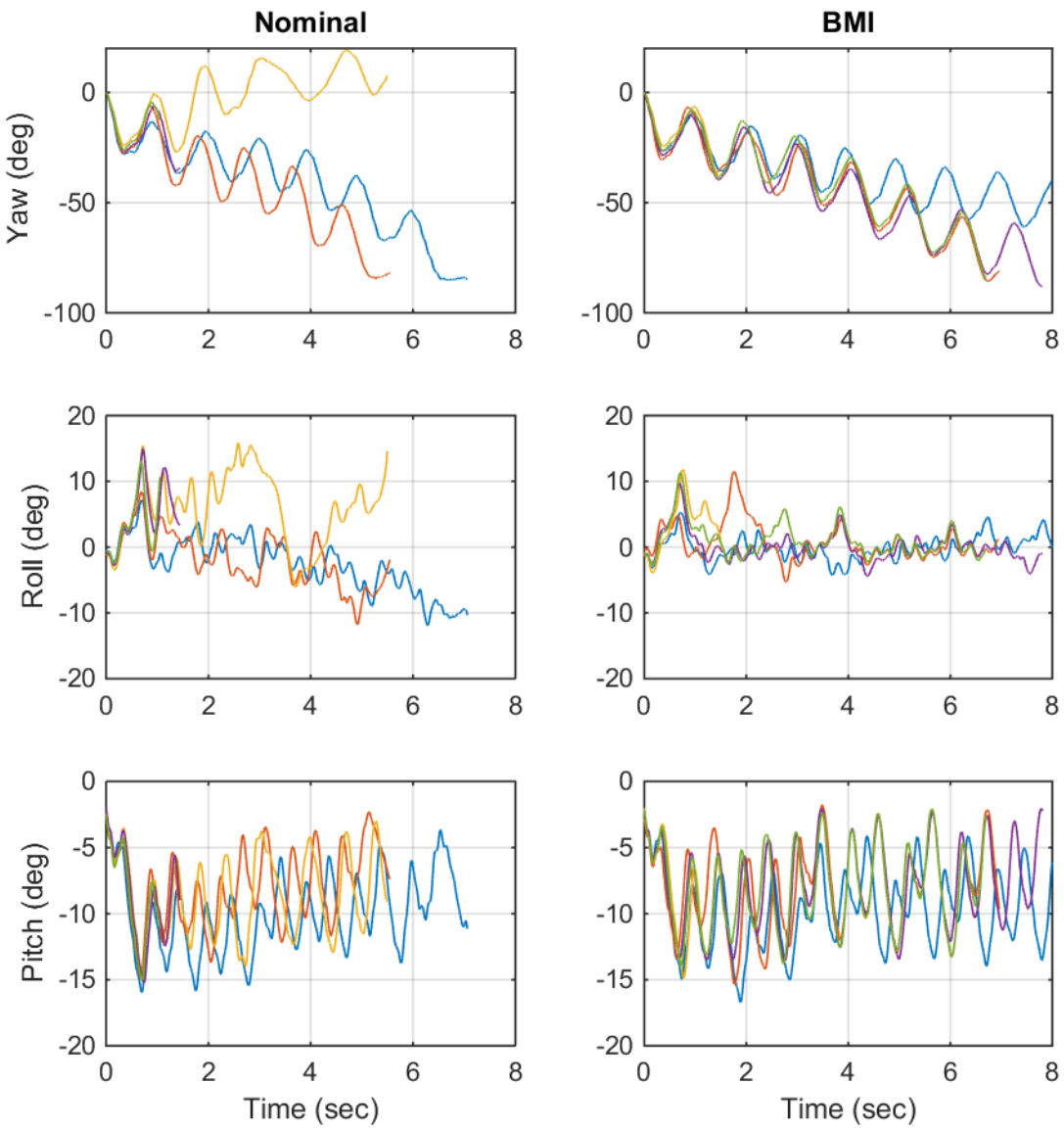


Figure 6.8: Torso Euler angles with event-based updates enabled. The plots compare the results from ten walking experiments, five of which used the nominal outputs (left column; experiments N1, N2, N3, N4, N5) and five of which used the optimized outputs (right column; experiments B1, B2, B3, B4, B5).

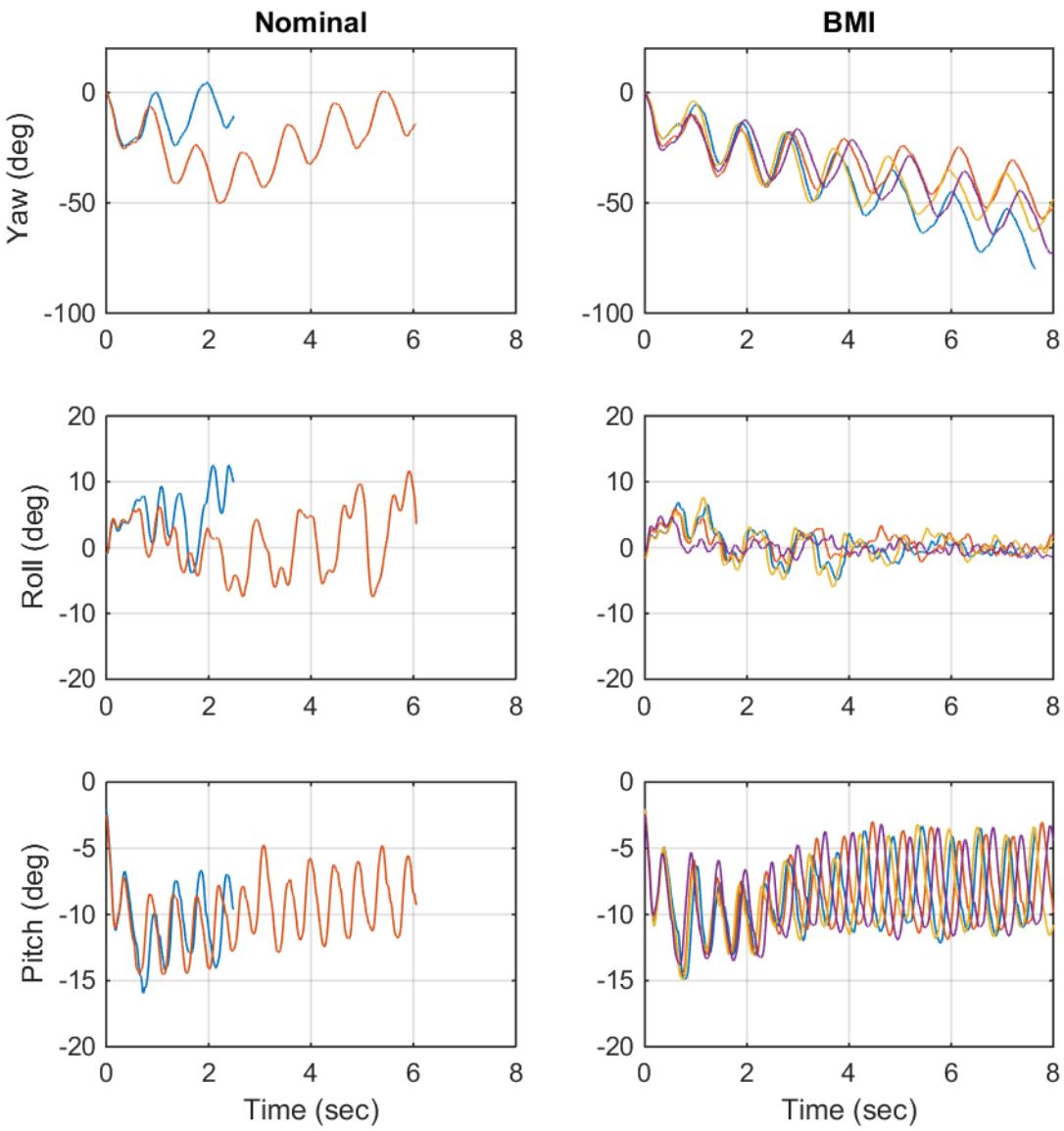


Figure 6.9: Torso Euler angles with event-based updates disabled. The plots compare the results from six walking experiments, two of which used the nominal outputs (left column; experiments N6, N7) and four of which used the optimized outputs (right column; experiments B6, B7, B8, B9).

step. The amplitude of the oscillation (between 6–10 degrees peak to peak) is somewhat larger than in simulation (5.5 degrees peak to peak; see Figure 6.3). The oscillation is more pronounced when using event-based updates (Figure 6.8) compared to experiments where the updates were disabled (Figure 6.9).

The most notable difference in the torso motion is in the roll angle. From the simulation, we expect the peak-to-peak torso roll to be about 4.4 degrees. After a transient following gait initiation, the optimized controller brings the torso oscillation to between 4 and 6 degrees peak to peak. The nominal controller fails to effectively stabilize the torso roll. These results do not change significantly when event-based updates are disabled.

To understand how the COM motion is affected by the different controllers, we plot in Figures 6.10 and 6.11 the linearized COM position with respect to the right foot.[3] From these plots we see that the relative motion of the COM in the sagittal plane is very similar for all four controllers tested. However, the motion of the COM in the lateral plane is quite exaggerated when the nominal outputs are used. When the optimized outputs are used, the COM is maintained very close to the nominal position.

The tracking errors were generally comparable. Figures 6.12 and 6.13 compare the desired evolutions with the actual trajectories of the controlled coordinates for experiments N1 and B1, respectively.

6.5.3 Discussion

These results suggest that the optimized virtual constraints are more effective at lateral stabilization than the nominal constraints. The robot walked farther, more consistently, and with less torso and COM oscillation in the lateral plane with the optimized virtual constraints compared to the nominal virtual constraints. We propose a mechanism by which

[3]Computed as $\hat{p}_{\mathrm{COM}}(q) = J_{\mathrm{COM}}\,(q - q_0)$ where $J_{\mathrm{COM}} = \frac{\partial p^{\mathrm{R}}_{\mathrm{COM}}}{\partial q}(q)\big|_{q=q_0}$ is a constant matrix, $p^{\mathrm{R}}_{\mathrm{COM}}(q)$ is the position of the COM with respect to the right leg, and q_0 is a symmetric, upright nominal configuration.

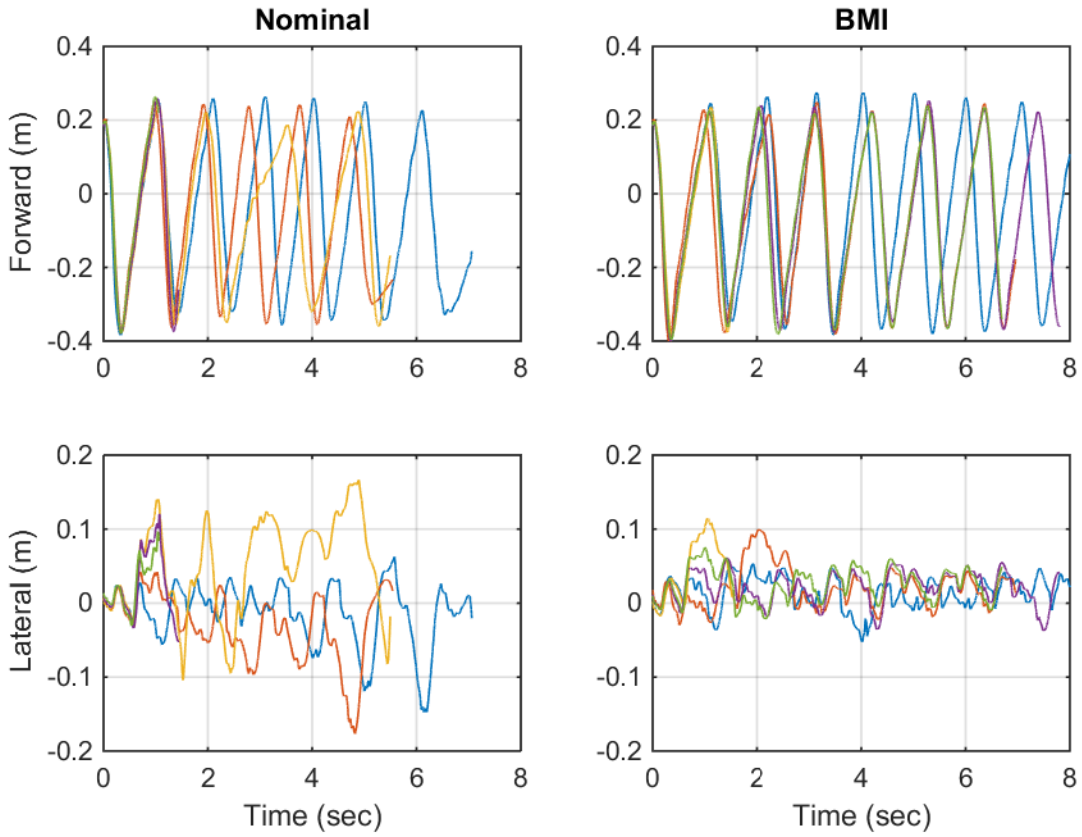


Figure 6.10: Linearized position of the COM with event-based updates enabled. The plots compare the results from ten walking experiments, five of which used the nominal outputs (left column; experiments N1, N2, N3, N4, N5) and five of which used the BMI-optimized outputs (right column; experiments B1, B2, B3, B4, B5).

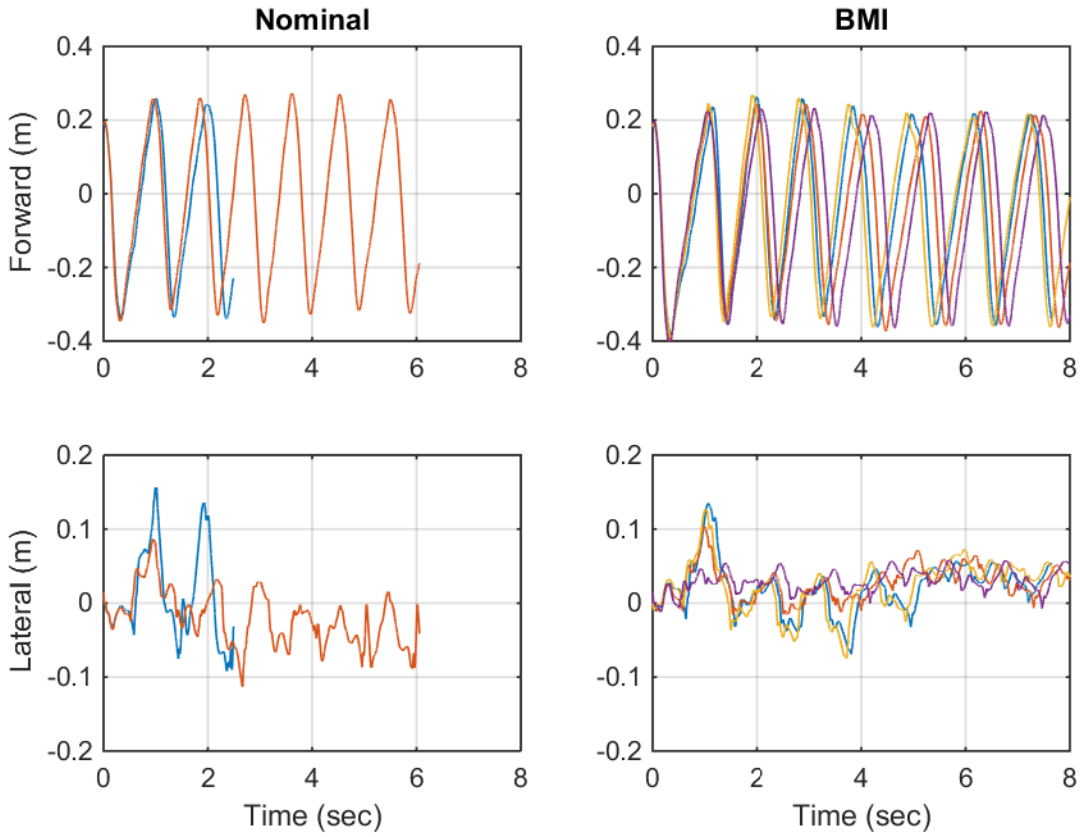


Figure 6.11: Linearized position of the COM with event-based updates disabled. The plots compare the results from six walking experiments, two of which used the nominal outputs (left column; experiments N6, N7) and four of which used the BMI-optimized outputs (right column; experiments B6, B7, B8, B9).

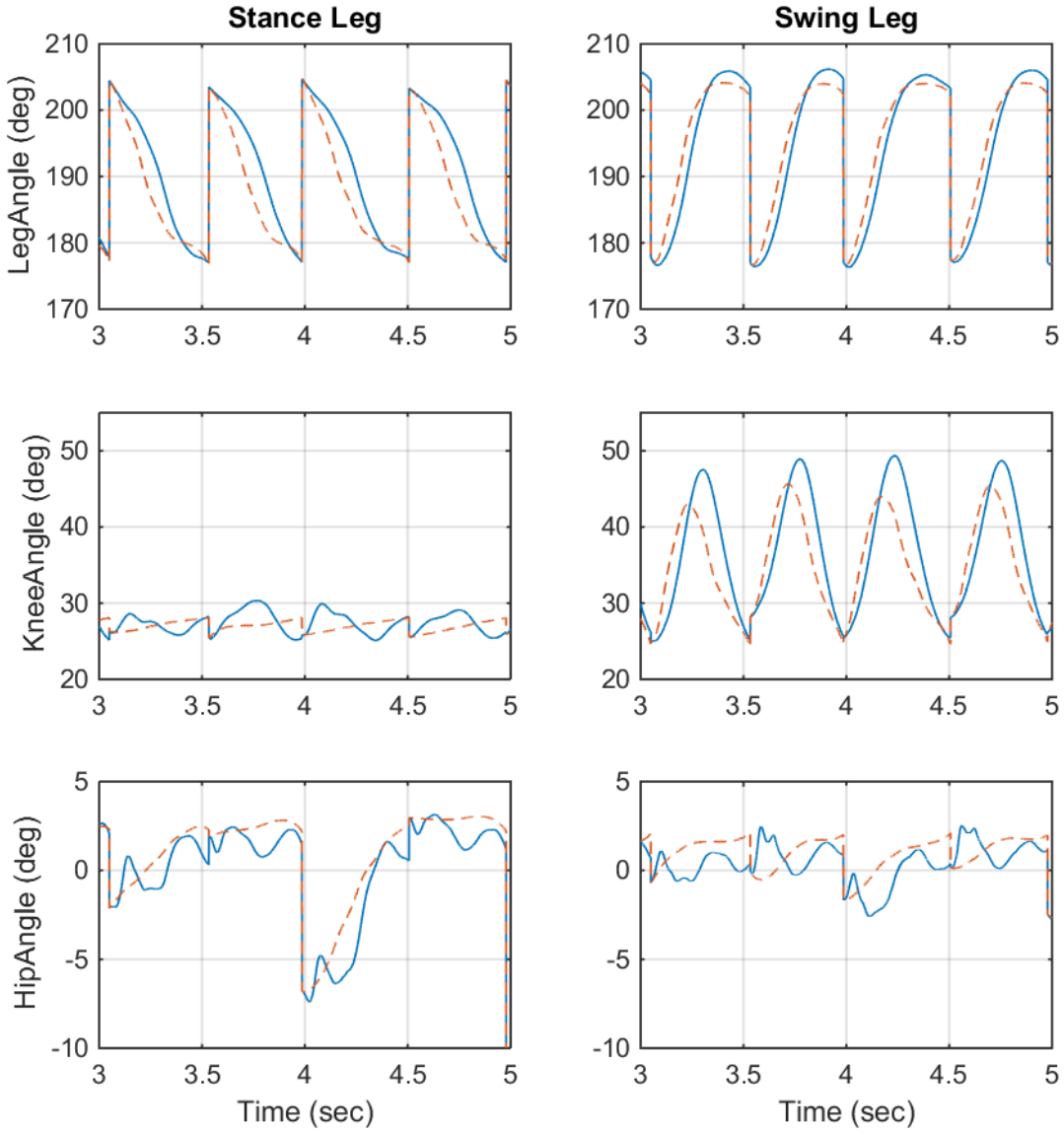


Figure 6.12: Tracking of desired evolutions when using the nominal outputs. The data are from experiment N1, and are representative of the other experiments. The dashed lines show the desired evolution of the controlled variables, and the solid lines represent their actual evolution.

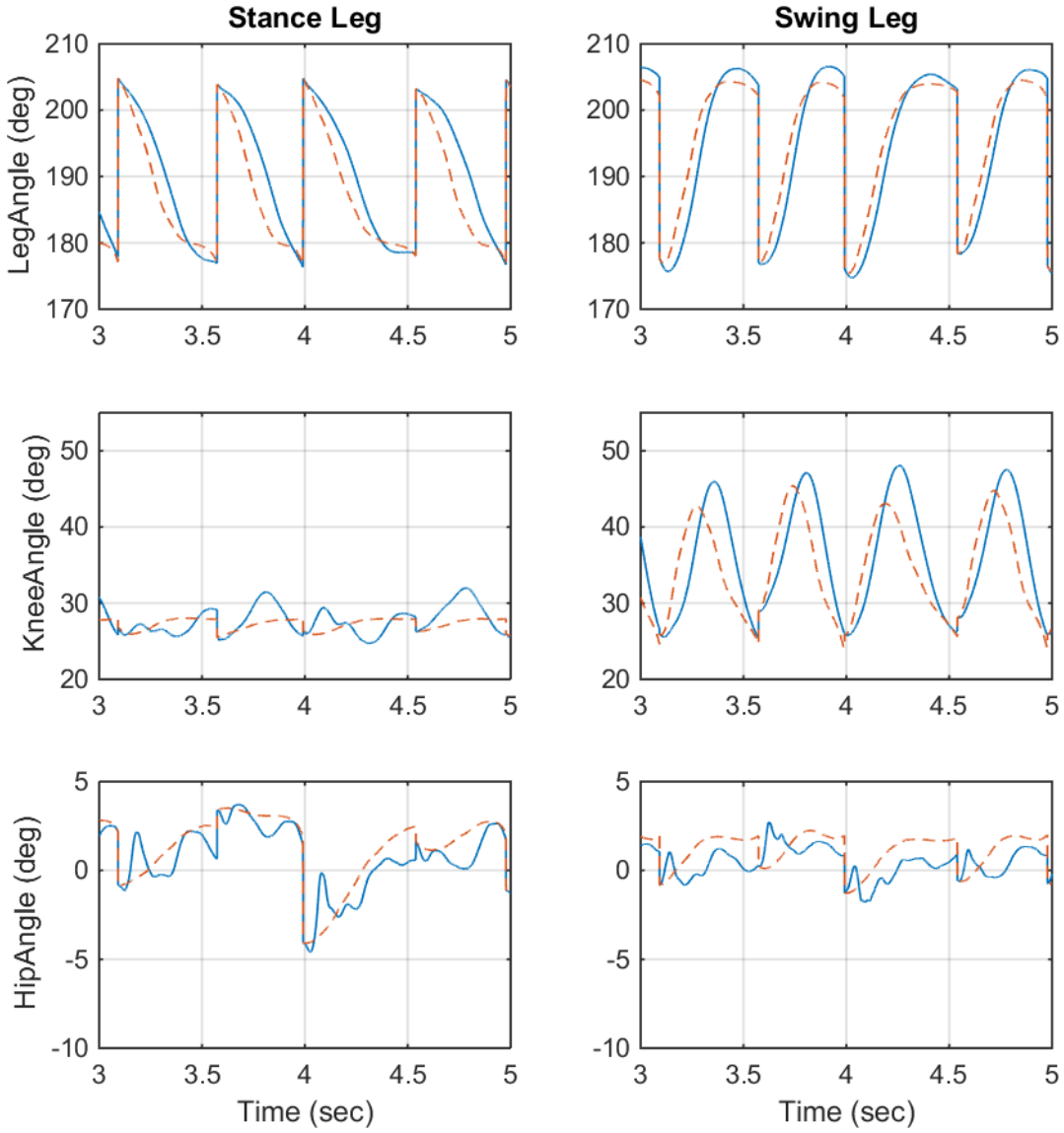


Figure 6.13: Tracking of desired evolutions when using the optimized outputs. The data are from experiment B1, and are representative of the other experiments. The dashed lines show the desired evolution of the controlled variables, and the solid lines represent their actual evolution.

this is achieved: In the lateral plane, the optimal perturbation $\Delta\xi$ primarily affects the swing hip angle. Furthermore, the perturbation is such that the swing foot moves in the direction the torso is rolling. This suggests that the optimized virtual constraints may improve foot placement in the lateral plane, thereby keeping the COM closer to the stance foot. This would reduce the tipping moment induced by gravity and thus reduce the magnitude of step-to-step oscillations in the lateral plane.

The alternative to stabilizing motion through foot placement is to couple the dynamics in such a way as to make *within-stride* adjustments to deviations from the desired orbit. Within-stride adjustments might include using internal angular momentum to keep the lateral position of the COM close to the stance foot, such as humans do when lunging or "windmilling" their appendages [10]. The only appendages with which MARLO could windmill to change its angular momentum in the lateral plane are the torso and the swing leg. However, the perturbation on the stance hip (which would affect lateral lunging with the torso) is very small, and the perturbation on the swing hip has the wrong sign to produce the desired reaction torque to stabilize the COM by windmilling.

Compared to the behavior in the lateral plane, the behavior of the robot in the sagittal plane under the two different sets of virtual constraints is surprisingly comparable. Given the size of the perturbation in the sagittal-plane virtual constraints, we might have expected a noticeable difference in the torso pitch, for example. It is possible that the actual effect of the perturbations in the sagittal-plane virtual constraints is negligible; an interesting follow-on to the experiments presented here would be to zero particular components of the perturbation and compare the results.

One apparent side effect of the optimized virtual constraints is the persistent yawing. While the robot tended to turn with both choices of virtual constraints, it did so more consistently and slightly more rapidly with the optimized virtual constraints. Likely contributing factors to the tendency to turn, in general, are asymmetry in the torso mass distribution

and asymmetry in the gait initiation. That yawing is more pronounced with the optimized virtual constraints is a consequence of the fact that we optimized the virtual constraints for stability *modulo yaw*.

While the results of these experiments are promising, we acknowledge several limitations. Due to the limited walking distance available in the lab, it is difficult to separate the effects of initial conditions from the long-term behavior of the robot under a particular controller. Variability in the initial conditions may be caused by small differences in how to robot is posed, how the robot initially falls forward, and where it takes its first step. More importantly, the robot itself may change over time. The toroidal feet, knee joints, and spring couplings may shift slightly from one experiment to the next. The batteries may also become partially depleted. These factors make a rigorous comparison between controllers more challenging.

CHAPTER 7

Conclusion

While much progress has been made toward the development of bipedal robots capable of operating in the real world, there are still significant gains to be made in terms of speed, agility, robustness, and energetic efficiency. The limit cycle walking paradigm offers a variety of ideas and methods that can help secure these gains by taking advantage of natural dynamics. In particular, limit cycle walking has demonstrated exceptional benefits in terms of energetic efficiency, and, embodied in the framework of hybrid zero dynamics, in terms of speed and robustness. A significant contribution of the theory of hybrid zero dynamics is to provide rigorous and systematic tools for gait design and stability analysis. These tools allow model-based design without the need to rely solely on simplified models such as the inverted pendulum. Their efficacy is demonstrated by the successful walking achieved on the rigid, planar robots Rabbit and ERNIE, and by the natural-looking running and robust walking achieved on the compliant, planar biped MABEL. Extending these planar results to the 3D setting will complement the ongoing work on humanoid walking to enable more robust, dynamic, and energetically efficient robot locomotion in the future.

Previous work with 3D simulation models had established the viability of HZD-based walking for 3D bipeds, but had also revealed an important difference with respect to the planar case with a single degree of underactuation: in the 3D case, the choice of virtual

constraints has a deciding effect on the stability of a periodic orbit. Furthermore, although choosing virtual constraints to ensure stability of an orbit can be a subtle problem, there was no systematic and scalable means of doing so. This thesis has addressed this problem by introducing an optimization-based method to choose virtual constraints. It has also made experimental contributions validating HZD-based control of 3D bipeds. These contributions are summarized below.

7.1 Summary of contributions

First experimental realization of 3D walking with virtual constraints. We have presented the first experimental realization of dynamic 3D walking using virtual constraints. As in earlier studies, which used HZD for 3D walking in simulation, the virtual constraints used in our initial experimental implementation are chosen based on intuition. Stability is achieved by constraining the lateral swing hip angle to be a function of the torso roll angle and of the lateral stance hip angle. This constraint implicitly adjusts the foot placement in a step-to-step manner. The virtual constraints also incorporate swing leg retraction, designed for robustness to external pushes with MARLO in a planar mode. The resulting controller is tested in the laboratory on the human-scale bipedal robot MARLO. The controller stabilizes the lateral motion for unassisted 3D walking at approximately 1 m/s. MARLO is one of only two known robots to walk in 3D with stilt-like feet.

Systematic selection of virtual constraints. To address the problem of choosing appropriate virtual constraints, we have introduced an optimization-based method to tune controller parameters for hybrid and non-hybrid systems. The method assumes that a parameterized feedback law generates a periodic orbit that does not depend on the controller parameters. Thus, while the Poincaré map (and, in particular, the eigenvalues of its Jacobian) may depend on the controller parameters, the fixed point does not. This allows

us to study a Taylor series approximation of the linearized Poincaré map and formulate an optimization problem to choose the controller parameters to stabilize its fixed point. Importantly, this method makes possible systematic stabilization without the need to recompute the linearized Poincaré map at each iteration of the optimization, making it applicable to systems of much greater dimension than previous methods. Furthermore, the optimization problem can also be formulated with a linear cost and BMI constraints. This relates the problem to other control problems which may be formulated as BMI optimization problems, and may provide additional flexibility in choosing numerical solvers.

One extension of the basic method allows a measure of disturbance rejection capability to be incorporated into the optimization. This makes it possible to choose controller parameters (such as virtual constraints) which are less sensitive to uncertainty in the switching manifold of the hybrid system. We also show how the optimization framework can be used in conjunction with the restricted Poincaré map. This extension allows the stabilization problem to be solved using the lower-dimensional hybrid zero dynamics, just as the gait design problem is solved. This may be advantageous for numerical reasons, since the lower dimension may accelerate the optimization process. Furthermore, as the designer may be able to understand the zero dynamics more readily than the full dynamics, optimizing the controller for the restricted system may facilitate interpretation of the solution. Finally, optimization of the restricted dynamics isolates the effect of the feedback law used to stabilize the zero dynamics manifold from the effect of the virtual constraints themselves, allowing these two components of the controller design to be carried out separately, if desired.

Experimental validation of BMI-based constraint design. Finally, we have demonstrated the systematic method for selecting virtual constraints to stabilize a walking gait for MARLO. A nominal choice of virtual constraints was parameterized in such a way as to allow each controlled coordinate to depend on the torso roll angle. We solved the BMI formulation of the stabilization problem to find the optimal parameters. The controller was

tested with MARLO walking on smaller, toroidal feet, which better match the point-foot model. The optimized controller leads to improved lateral control compared to the nominal virtual constraints.

7.2 Future directions

Systematic selection of the phase variable. The optimization-based method for designing controllers opens up many new possibilities. In this work we have used the optimization to select controlled variables for walking. Another possibility is to use the optimization to select a mechanical phase variable. The phase variable not only serves as a way of parameterizing time along a periodic orbit, but, much more importantly, *partitions the state space* into equivalence classes (the level sets of θ). To each equivalence class, the controller associates a single vector of desired outputs $h_d(\theta)$. Thus a poorly chosen mechanical phase variable might group into the same equivalence class points of the state space where the robot behavior is very different. This would have important consequences for stability and disturbance rejection, hence the motivation for making the choice of phase variable systematic.

One way to parameterize the phase variable for optimization is analogous to the way we parameterized the controlled variables: as a linear combination of the configuration q. However, this could lead to difficulty when the solver chooses a linear combination which is not monotonic along the orbit. As an alternative, one could form a collection of any number of arbitrary functions of the configuration which are known to be monotonic along the desired orbit, then parameterize the phase variable as a convex combination of these functions.

Robustness and disturbance rejection. The optimizations described for stabilization and disturbance rejection could be augmented with additional performance criteria. For

example, we noted that our disturbance rejection metric is quite similar to Hobbelen and Wisse's gait sensitivity norm. In fact, one could easily incorporate the gait sensitivity norm itself as part of the cost in the NLP.

An important area for continued research is in robustness of the optimized controllers. We noted briefly that the optimization method extends trivially to simultaneous stabilization of a collection of orbits. These orbits could all be for the same hybrid system, or they could be orbits for different hybrid systems. It would be interesting to find a collection of "similar" orbits for a collection of hybrid systems which differ in their parameters, and to look for a controller which stabilizes all of the orbits. This has the flavor of a robust control problem, but rigorously proving the robustness of the resulting controller for a wider collection of hybrid models may be very challenging.

Walking in place. In all of the 3D experiments we have described, a key challenge was starting from a static pose and guiding the robot toward the periodic orbit. An alternative to the gait initiation process we have used (which required considerable hand tuning) is to let the robot first walk in place, then slowly increase its forward momentum over multiple steps until it is close to the periodic orbit. In fact, walking in place is interesting in its own right. We have begun work toward an in-place walking controller for MARLO, which is described in Appendix A. The gait design uses HZD with a phase variable defined in the lateral plane. Because the phase variable is not monotonic during each stance phase, the stance phase is split into two sub-phases. Using this lateral phase variable, we designed an in-place walking gait. However, optimizing the controlled variables to stabilize the gait as we did for regular walking required excessively large perturbations $\Delta\xi$. We hypothesize that this is because the parametrization of the virtual constraints was not sufficiently flexible. An interesting problem would be to determine a more appropriate parametrization or a different class of virtual constraints which would simplify stabilization of in-place walking.

APPENDICES

APPENDIX A

Walking in place

Walking in place is important both theoretically and practically. As a 3D point-foot robot cannot stand still, the ability to walk in place could serve as a useful starting stage for 3D experiments with point feet. It is interesting theoretically because it introduces a new challenge in control design. Application of HZD relies on the notion of a mechanical phase variable. In forward walking there are various choices, including the virtual stance leg angle, the horizontal position of the center of mass, and the horizontal position of the center of pressure [98]. Walking in place differs from forward walking in the sense that there is not an obvious choice of timing variable which is monotonic throughout each support phase. In this chapter we show how this problem may be addressed, and present a controller design for stable in place walking with MARLO.

In the next section we describe a hybrid controller for walking in place on point feet. In the controller design, we do not worry about slow yawing motion; in fact, if the yaw can be controlled, then this may prove useful for turning in place. Similarly, position is not actively regulated; the goal is simply to stay up without travelling in any particular direction very quickly.

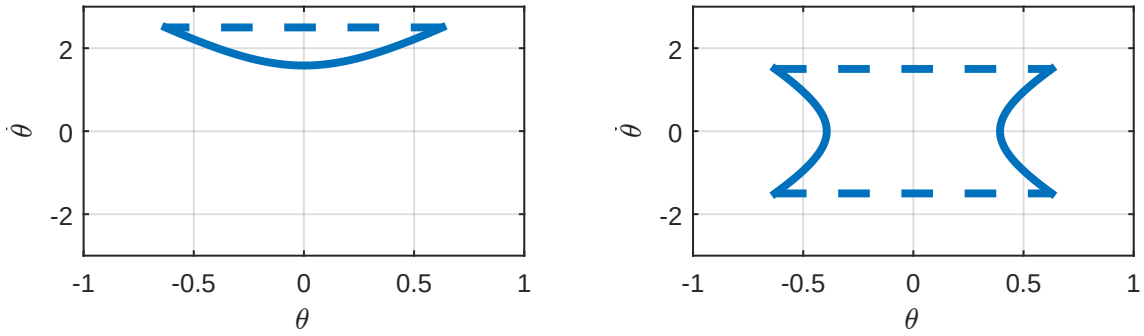


Figure A.1: Phase portraits for decoupled inverted pendulum model of walking. The motion in the sagittal plane (a) is identical for left and right stance, while in the lateral plane (b) the motion is mirrored about the vertical.

Gait design

Phase variable for walking in place

We have already noted that (forward) bipedal walking can be studied using inverted pendulum models. The motion of the center of mass in the sagittal plane is modeled by a pendulum which moves monotonically through the upright position, as shown in Fig. A.1a. In the lateral plane, the center of mass initially moves toward the upright position, but it stops short of the vertical and reverses direction in the latter half of the step (Fig. A.1b).

Walking in place may be considered a limiting case of forward walking as the stride length approaches zero. In the basic inverted pendulum model, this corresponds to fixing the angle of the center of mass in the sagittal place at the vertical; the curve in Fig. A.1a would then collapse to a point at the origin. (In models with additional dynamics, such as an inverted pendulum with a flywheel [10] or a bipedal robot, small periodic motions in the sagittal plane might also result in periodic walking in place.) It is critical that the lateral velocity of the center of mass changes sign strictly before the inverted pendulum reaches the vertical, otherwise it will fall to the outside instead of rocking back toward the swing foot.

Since walking in place is dominated by the lateral rocking motion, a natural choice for the phase variable is the angle of the inverted pendulum in the lateral plane. But as this angle is not monotonic within each stance phase (see Fig. A.1b), we cannot use it to parameterize arbitrary virtual constraints. To illustrate what this means, suppose we are given a periodic trajectory $x^*(t), t = [0, T]$ corresponding to an in-place walking gait. Then for a particular choice of controlled coordinates $h_0(q)$, we might wish to determine desired evolutions $h_d(\theta)$ such that $x^*(t)$ lies in the zero dynamics manifold $\mathcal{Z} = \{x = (q, \dot{q}) | h(x) = \mathcal{L}_f h(x) = 0\}$, where $h(q) = h_0(q) - h_d(\theta)$. In other words, we want to find $h_d(\theta)$ which satisfies

$$h_0(q^*(t)) - h_d(\theta(q^*(t))) \quad \forall\ t \in [0, T]. \tag{A.1}$$

Unless the controlled coordinates on the periodic orbit $h_0(q^*(t))$ are symmetric in θ, this will not be possible.

Since we do not wish to restrict our search to orbits satisfying (A.1), we have to parameterize the desired evolutions differently. One way to do this is to split the stance phase into two distinct segments (which we will call "early" and "late") such that θ is monotonic on each segment. We then let the desired evolutions be given by

$$h_d(\theta) = \begin{cases} h_d^1(\theta), & \dot{\theta} \geq 0 \\ h_d^2(-\theta), & \dot{\theta} < 0. \end{cases} \tag{A.2}$$

It is worth mentioning a few consequences of this choice.

First, we have effectively introduced an additional continuous phase into our hybrid model. The open-loop dynamics $\dot{x} = f(x) + g(x)u$ of the system in the "early" and "late" stance phases are identical; however, the parameterized feedback law $u = \Gamma(x, \alpha)$, the guard set, and the discrete transition are different.

Second, the existence of the zero dynamics is no longer guaranteed. To verify that the zero dynamics exists for a given periodic orbit and choice of virtual constraints, we must check *a posteriori* the invertibility of the matrix D_{zero} along the periodic orbit.

Third, we note that (A.2) still imposes a constraint on the periodic orbit: since $\dot{h}_d(\theta) = 0$ whenever $\dot{\theta} = 0$, the velocities of controlled coordinates must all simultaneously vanish at the instant the rocking motion reverses direction. An interesting corollary is that, as long as we choose h_d^1 and h_d^2 so that the desired evolutions h_d are *continuous* at the direction reversal, it is guaranteed that h_d will be *continuously differentiable* at that point.

Finally, without imposing additional constraints on h_d^1 and h_d^2, the control torques may be discontinuous at the direction reversal. This is undesirable, and will be handled in the controller design.

Motivated by the inverted pendulum model, we select as the phase variable θ a linear approximation of MARLO's position of the center of mass in the lateral plane. Specifically, we set

$$\theta(q) = -1.04q_{yT} - 0.967q_{3R} - 0.0154q_{3L}. \tag{A.3}$$

Gait design

The generalized coordinates q and velocities $\dot{q}$ are continuous at the direction reversal. We wish to ensure that the control torques u (and hence the accelerations $\ddot{q}$ are also continuous. From (2.11) we have that the torques along the periodic orbit corresponding to the virtual constraints $0 = h(q) = h_0(q) - h_d(\theta)$ are

$$\begin{aligned} u^* &= \left(\frac{\partial h}{\partial q}D^{-1}B\right)^{-1}\left[\frac{\partial h}{\partial q}D^{-1}H - \frac{\partial}{\partial q}\left(\frac{\partial h}{\partial q}\dot{q}\right)\dot{q}\right] \\ &= \left(\frac{\partial h}{\partial q}D^{-1}B\right)^{-1}\left[\frac{\partial h}{\partial q}D^{-1}H - \frac{\partial}{\partial q}\left(\frac{\partial h}{\partial q}\dot{q}\right)\dot{q}\right] \end{aligned}$$

If the controlled coordinates h_0 are defined equivalently in the two phases, then to ensure equality of u^- and u^+ it is sufficient that

$$\frac{\partial h_d^1}{\partial \theta} = \frac{\partial h_d^2}{\partial \theta}. \tag{A.4}$$

at the mid-stance transition.

If h_d^1 and h_d^2 are chosen to be Bézier polynomials in s, where $s = \frac{\theta - \theta^+}{\theta^- - \theta^+}$ is normalized by the initial and final values of θ in each phase, then (A.4) is equivalent to

$$(\alpha_M^1 - \alpha_{M-1}^1)\frac{1}{\theta_1^- - \theta_1^+} = (\alpha_1^2 - \alpha_0^2)\frac{1}{\theta_2^- - \theta_2^+}. \tag{A.5}$$

An optimization problem was defined to minimize the norm of the equality constraint violation subject to constraints similar to those listed previously for forward walking (see Table 6.1).

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] R. Alexander, "Three uses for springs in legged locomotion," *International Journal of Robotics Research*, vol. 9, no. 2, pp. 53–61, 1990.

[2] V. A. Tucker, "The energetic cost of moving about: walking and running are extremely inefficient forms of locomotion. Much greater efficiency is achieved by birds, fish—and bicyclists," *American Scientist*, pp. 413–419, 1975.

[3] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, Y. Kawai, F. Tomita, and H. Hirukawa, "Cooperative works by a human and a humanoid robot," in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 3, Sept 2003, pp. 2985–2991 vol.3.

[4] D. Hobbelen, T. de Boer, and M. Wisse, "System overview of bipedal robots Flame and TUlip: Tailor-made for limit cycle walking," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, Sept 2008, pp. 2486–2491.

[5] K. Tanie, "Humanoid robot and its application possibility," in *Multisensor Fusion and Integration for Intelligent Systems, MFI2003. Proceedings of IEEE International Conference on*, July 2003, pp. 213–214.

[6] P. Hebert, M. Bajracharya, J. Ma, N. Hudson, A. Aydemir, J. Reid, C. Bergh, J. Borders, M. Frost, and M. Hagman, "Mobile manipulation and mobility as manipulation: Design and algorithms of RoboSimian," DTIC Document, Tech. Rep., 2014.

[7] A. A. Frank, "Automatic control systems for legged locomotion machines," University of Southern California Los Angeles, Tech. Rep., 1968.

[8] T. Koolen, T. de Boer, J. Rebula, A. Goswami, and J. Pratt, "Capturability-based analysis and control of legged locomotion, Part 1: Theory and application to three simple gait models," *The International Journal of Robotics Research*, vol. 31, no. 9, pp. 1094–1113, Jul. 2012.

[9] S. H. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, no. 307, pp. 1082–1085, 2005.

[10] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture Point: A Step toward Humanoid Push Recovery," *2006 6th IEEE-RAS International Conference on Humanoid Robots*, 2006.

[11] H.-W. Park, K. Sreenath, A. Ramezani, and J. Grizzle, "Switching control design for accommodating large step-down disturbances in bipedal robot walking," in *Int. Conf. on Robotics and Automation (ICRA)*, May 2012.

[12] G. Nelson, A. Saunders, N. Neville, B. Swilling, J. Bondaryk, D. Billings, C. Lee, R. Playter, and M. Raibert, "PETMAN: A humanoid robot for testing chemical protective clothing," *Journal of the Robotics Society of Japan*, vol. 30, no. 4, pp. 372–377, 2012.

[13] P. A. Bhounsule, J. Cortell, A. Grewal, B. Hendriksen, J. G. D. Karssen, C. Paul, and A. Ruina, "Low-bandwidth reflex-based control for lower power walking: 65 km on a single battery charge," *The International Journal of Robotics Research*, vol. 33, no. 10, pp. 1305–1321, 2014.

[14] D. Winter, "Human balance and posture control during standing and walking," *Gait & Posture*, vol. 3, no. 4, pp. 193 – 214, 1995.

[15] J.-Y. Kim, I.-W. Park, and J.-H. Oh, "Experimental realization of dynamic walking of the biped humanoid robot KHR-2 using zero moment point feedback and inertial measurement," *Advanced Robotics*, vol. 20, no. 6, pp. 707–736, Jan. 2006.

[16] H. Yokoi, Kazuhito and Kanehiro, Fumio and Kaneko, Kenji and Fujiwara, Kiyoshi and Kajita, Shuji and Hirukawa, "Experimental Study of Biped Locomotion of Humanoid Robot HRP-1S," in *Experimental Robotics VIII*, ser. Springer Tracts in Advanced Robotics, B. Siciliano and P. Dario, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, Jun. 2003, vol. 5, ch. Experiment, pp. 75–84.

[17] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenake, "The development of Honda humanoid robot," in *Proc. of the IEEE International Conference on Robotics and Automation, Leuven, Belgium*, May 1998, pp. 1321–1326.

[18] T. McGeer, "Passive dynamic walking," *International Journal of Robotics Research*, vol. 9, no. 2, pp. 62–82, 1990.

[19] D. G. Hobbelen and M. Wisse, *Limit Cycle Walking, Humanoid Robots, Human-like Machines*. i-Tech Education and Publishing, 2007, ch. 14, pp. 277–294.

[20] IHMC Robotics Lab, "Atlas walking over randomness," Online: http://www.youtube.com/watch?v=d4qqM7_E11s, Nov 2013.

[21] A. Takanishi, G. Naito, M. Ishida, and I. Kato, "Realization of plane walking by the biped walking robot WL-10R," in *Theory And Practice of Robots And Manipulators: Proceedings of RoManSy '84, the Fifth CISM-IFToMM Symposium*, A. Morecki, G. Bianchi, and K. Kedzior, Eds.

[22] S. O. Anderson, M. Wisse, C. G. Atkeson, J. K. Hodgins, G. J. Zeglin, and B. Moyer, "Powered bipeds based on passive dynamic principles," in *5th IEEE/RAS International Conference on Humanoid Robots*, 2005, pp. 110–116.

[23] M. Vukobratovic and B. Borovac, "Zero-moment point âĂŤ thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 01, no. 01, pp. 157–173, 2004.

[24] A. Goswami, "Postural stability of biped robots and the foot-rotation indicator (FRI) point," *International Journal of Robotics Research*, vol. 18, no. 6, pp. 523–33, Jun. 1999.

[25] H.-o. Lim and A. Takanishi, "Biped walking robots created at Waseda University: WL and WABIAN family." *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, vol. 365, pp. 49–64, 2007.

[26] R. Sellaouti, O. Stasse, S. Kajita, K. Yokoi, and A. Kheddar, "Faster and smoother walking of humanoid hrp-2 with passive toe joints," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, Oct 2006, pp. 4909–4914.

[27] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi, and K. Tanie, "Planning walking patterns for a biped robot," *Robotics and Automation, IEEE Transactions on*, vol. 17, no. 3, pp. 280–289, Jun 2001.

[28] J. Pratt, T. Koolen, T. de Boer, J. Rebula, S. Cotton, J. Carff, M. Johnson, and P. Neuhaus, "Capturability-based analysis and control of legged locomotion, Part 2: Application to M2V2, a lower-body humanoid," *The International Journal of Robotics Research*, vol. 31, no. 10, pp. 1117–1133, Aug. 2012.

[29] M. Missura and S. Behnke, "Lateral capture steps for bipedal walking," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*. IEEE, Oct. 2011, pp. 401–408.

[30] 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, pp. 51–64, January 2001.

[31] 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.

[32] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*, ser. Control and Automation. Boca Raton, FL: CRC Press, June 2007.

[33] 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 *Proceedings of the IEEE International Conference on Decision and Control*, Seville, Spain, Dec. 2005, pp. 4199–4206.

[34] ——, "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.

[35] A. Martin, D. Post, and J. Schmiedeler, "Design and experimental implementation of a hybrid zero dynamics controller for planar bipeds with curved feet."

[36] K. Sreenath, H.-W. Park, I. Poulakakis, and J. Grizzle, "Design and experimental implementation of a compliant hybrid zero dynamics controller for walking on MABEL," in *IEEE Conference on Decision and Control*, 2010, pp. 280–287.

[37] K. Sreenath, H.-W. Park, and J. Grizzle, "Design and experimental implementation of a compliant hybrid zero dynamics controller with active force control for running on MABEL," in *Int. Conf. on Robotics and Automation (ICRA)*, May 2012.

[38] T. Yang, E. R. Westervelt, A. Serrani, and J. P. Schmiedeler, "A framework for the control of stable aperiodic walking in underactuated planar bipeds," *Autonomous Robots*, vol. 27, no. 3, pp. 277–290, 2009.

[39] H. W. Park, "Control of a bipedal robot walker on rough terrain," Ph.D. dissertation, University of Michigan, 2012.

[40] B. Griffin and J. Grizzle, "Nonholonomic virtual constraints for dynamic walking," in *Submitted on 24 Mar. 2015, 54th IEEE Conference of Decision and Control (CDC 2015)*, 2015.

[41] A. D. Ames and R. D. Gregg, "Stably extending two-dimensional bipedal walking to three," in *American Control Conference*, New York, U.S.A., Jul. 2007, pp. 2848–2854.

[42] C. Chevallereau, J. Grizzle, and C. 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.

[43] C.-L. Shih, J. Grizzle, and C. Chevallereau, "From Stable Walking to Steering of a 3D Bipedal Robot with Passive Point Feet," *Robotica*, vol. 30, no. 07, pp. 1119–1130, December 2012.

[44] A. D. Ames, E. A. Cousineau, and M. J. Powell, "Dynamically stable robotic walking with NAO via human-inspired hybrid zero dynamics," in *Hybrid Systems, Computation and Control (HSCC)*, Philadelphia, April 2012.

[45] A. Ramezani, J. W. Hurst, K. Akbari Hamed, and J. W. Grizzle, "Performance Analysis and Feedback Control of ATRIAS, A Three-Dimensional Bipedal Robot," *Journal of Dynamic Systems, Measurement, and Control*, vol. 136, no. 2, Dec. 2013.

[46] A. Ramezani, "Feedback control design for MARLO, a 3D-bipedal robot," PhD Thesis, University of Michigan, 2013.

[47] 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, April 1998.

[48] M. Spong and F. Bullo, "Controlled symmetries and passive walking," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 1025–1031, July 2005.

[49] S. H. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, pp. 1082–85, 2005.

[50] W. M. Bouwman, G. van Oort, E. C. Dertien, J. F. Broenink, and R. Carloni, "Dynamic walking stability of the tulip robot by means of the extrapolated center of mass," 2010.

[51] M. H. Raibert, "Hopping in legged systems—modeling and simulation for the two-dimensional one-legged case," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 14, no. 3, pp. 451–63, Jun. 1984.

[52] M. Raibert and J. Brown, H., "Experiments in balance with a 2D one-legged hopping machine," *ASME Journal of Dynamic Systems Measurement, and Control*, vol. 106, pp. 75–81, 1984.

[53] M. H. Raibert, H. B. Brown, and M. Chepponis, "Experiments in Balance with a 3D One-Legged Hopping Machine," *The International Journal of Robotics Research*, vol. 3, no. 2, pp. 75–92, Jun. 1984.

[54] M. H. Raibert, M. Chepponis, and J. Brown, H., "Running on four legs as though they were one," *IEEE Transactions on Robotics and Automation*, vol. RA-2, no. 2, pp. 70–82, Jun. 1986.

[55] F. Peuker, C. Maufroy, and A. Seyfarth, "Leg-adjustment strategies for stable running in three dimensions." *Bioinspiration & biomimetics*, vol. 7, no. 3, pp. 1–13, Sep. 2012.

[56] M. A. Sharbafi, A. Seyfarth, M. N. Ahmadabadi, and M. J. Yazdanpanah, "Novel leg adjustment approach for hopping and running," in *Dynamic Walking Conference*, Carnegie Mellon University, Pittsburgh, Pennsylvania, June 2013.

[57] A. Seyfarth, H. Geyer, and H. Herr, "Swing leg retraction: A simple control model for stable running," *Journal of Experimental Biology*, vol. 206, pp. 2547–2555, 2003.

[58] S. J. Hasaneini, C. J. B. Macnab, J. E. A. Bertram, and A. Ruina, "Seven reasons to brake leg swing just before heel strike," in *Dynamic Walking 2013*. Carnegie Mellon University, Pittsburgh, Pennsylvania: Dynamic Walking, June 2013.

[59] H.-M. Maus, S. W. Lipfert, M. Gross, J. Rummel, and A. Seyfarth, "Upright human gait did not provide a major mechanical challenge for our ancestors." *Nature Communications*, vol. 1, no. 6, p. 70, Jan. 2010.

[60] J. Pratt, P. Dilworth, and G. Pratt, "Virtual model control of a bipedal walking robot," in *Proc. of the IEEE International Conference on Robotics and Automation, Albuquerque, N.M.*, April 1997, pp. 193–198.

[61] J. Pratt, M. Chee, A. Torres, P. Dilworth, and G. Pratt, "Virtual model control: an intuitive approach for bipedal locomotion," *International Journal of Robotics Research*, vol. 20, no. 2, pp. 129–143, February 2001.

[62] K. Yin, K. Loken, and M. van de Panne, "SIMBICON: Simple biped locomotion control," *ACM Transactions on Graphics*, vol. 26, no. 3, 2007.

[63] S. Coros, A. Karpathy, B. Jones, L. Reveret, and M. van de Panne, "Locomotion skills for simulated quadrupeds," *ACM Transactions on Graphics*, vol. 30, no. 4, pp. 59:1–59:12, 2011.

[64] C. Gehring, S. Coros, M. Hutter, M. Bloesch, M. Hoepflinger, and R. Siegwart, "Control of dynamic gaits for a quadrupedal robot," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013, pp. 3287–3292.

[65] M. Diehl, K. Mombaur, and D. Noll, "Stability optimization of hybrid periodic systems via a smooth criterion," *Automatic Control, IEEE Transactions on*, vol. 54, no. 8, pp. 1875–1880, Aug 2009.

[66] A. S. Shiriaev, L. B. Freidovich, and I. R. Manchester, "Can we make a robot ballerina perform a pirouette? orbital stabilization of periodic motions of underactuated mechanical systems," *Annual Reviews in Control*, vol. 32, no. 2, pp. 200–211, 2008.

[67] A. S. Shiriaev and L. B. Freidovich, "Transverse linearization for impulsive mechanical systems with one passive link," *Automatic Control, IEEE Transactions on*, vol. 54, no. 12, pp. 2882–2888, 2009.

[68] A. S. Shiriaev, L. B. Freidovich, and S. V. Gusev, "Transverse linearization for controlled mechanical systems with several passive degrees of freedom," *Automatic Control, IEEE Transactions on*, vol. 55, no. 4, pp. 893–906, 2010.

[69] I. R. Manchester, U. Mettin, F. Iida, and R. Tedrake, "Stable dynamic walking over uneven terrain," *The International Journal of Robotics Research*, vol. 30, no. 3, pp. 265–279, 2011.

[70] K. Byl and R. Tedrake, "Metastable walking machines," *The International Journal of Robotics Research*, vol. 28, no. 8, pp. 1040–1064, 2009.

[71] C. O. Saglam and K. Byl, "Quantifying the trade-offs between stability versus energy use for underactuated biped walking," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 2550–2557.

[72] D. G. E. Hobbelen and M. Wisse, "A disturbance rejection measure for limit cycle walkers: The gait sensitivity norm," *Robotics, IEEE Transactions on*, vol. 23, no. 6, pp. 1213–1224, 2007.

[73] B. Griffin and J. Grizzle, "Walking gait optimization for accommodation of unknown terrain height variations," 2015.

[74] T. S. Parker and L. O. Chua, *Practical Numerical Algorithms for Chaotic Systems*. New York, NY: Springer-Verlag, 1989.

[75] V. Guillemin and A. Pollack, *Differential Topology*, ser. AMS Chelsea Publishing Series. AMS Chelsea Pub., 2010.

[76] 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.

[77] J. H. Choi and J. W. Grizzle, "Planar bipedal walking with foot rotation," 2005, pp. 4909–16.

[78] K. Akbari Hamed, N. Sadati, W. A. Gruver, and G. A. Dumont, "Stabilization of Periodic Orbits for Planar Walking With Noninstantaneous Double-Support Phase," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 42, no. 3, pp. 685–706, May 2012.

[79] Y. Hurmuzlu and T. Chang, "Rigid body collisions of a special class of planar kinematic chains," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 22, no. 5, pp. 964–971, 1992.

[80] F. Plestan, J. W. Grizzle, E. R. Westervelt, and G. Abba, "Stable walking of a 7-DOF biped robot," *IEEE Transactions on Robotics*, vol. 19, no. 4, pp. 653–668, August 2003.

[81] 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, 1995.

[82] 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 (NOLCOS)*, Bologna, Italy, September 2010.

[83] B. J. Stephens and C. G. Atkeson, "Dynamic Balance Force Control for compliant humanoid robots," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Oct. 2010, pp. 1248–1255.

[84] K. Sreenath, H. Park, I. Poulakakis, and J. Grizzle, "A compliant hybrid zero dynamics controller for stable, efficient and fast bipedal walking on MABEL," *International Journal of Robotics Research*, vol. 30(9), pp. 1170–1193, 2011.

[85] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ: Prentice Hall, 2002.

[86] M. W. Spong, "Modeling and control of elastic joint robots," *Journal of Dynamic Systems, Measurement, and Control*, vol. 109, pp. 310–9, Dec. 1987.

[87] T. Geijtenbeek, M. van de Panne, and A. F. van der Stappen, "Flexible muscle-based locomotion for bipedal creatures," *ACM Transactions on Graphics*, vol. 32, no. 6, 2013.

[88] J. W. Grizzle. Dynamic Leg Locomotion. Youtube Channel: http://www.youtube.com/DynamicLegLocomotion.

[89] K. Akbari Hamed, B. G. Buss, and J. W. Grizzle, "Exponentially stabilizing continuous-time controllers for periodic orbits of hybrid systems: Application to bipedal locomotion with ground height variations," *IJRR*, submitted 20 Jan 2015.

[90] ——, "Continuous-time controllers for stabilizing periodic orbits of hybrid systems: Application to an underactuated 3d bipedal robot," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, Dec 2014, pp. 1507–1513.

[91] A. Isidori, *Nonlinear Control Systems*, 3rd ed. Berlin: Springer-Verlag, 1995.

[92] J. G. VanAntwerp and R. D. Braatz, "A tutorial on linear and bilinear matrix inequalities," *Journal of Process Control*, vol. 10, no. 4, pp. 363–385, 2000.

[93] M. Safonov, K. Goh, and J. Ly, "Control system synthesis via bilinear matrix inequalities," in *American Control Conference, 1994*, vol. 1, June 1994, pp. 45–49 vol.1.

[94] O. Toker and H. Ozbay, "On the np-hardness of solving bilinear matrix inequalities and simultaneous stabilization with static output feedback," in *American Control Conference, Proceedings of the 1995*, vol. 4, Jun 1995, pp. 2525–2526 vol.4.

[95] A. Hassibi, J. How, and S. Boyd, "A path-following method for solving bmi problems in control," in *American Control Conference, 1999. Proceedings of the 1999*, vol. 2, Jun 1999, pp. 1385–1389 vol.2.

[96] M. Kočvara and M. Stingl, "Pennon: A code for convex nonlinear and semidefinite programming," *Optimization methods and software*, vol. 18, no. 3, pp. 317–333, 2003.

[97] J. Lofberg, "Yalmip : a toolbox for modeling and optimization in matlab," in *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, Sept 2004, pp. 284–289.

[98] R. D. Gregg, E. J. Rouse, L. J. Hargrove, and J. W. Sensinger, "Evidence for a time-invariant phase variable in human ankle control," *PLoS ONE*, vol. 9, p. e89163, 2014.